ELSEVIER

Neurocomputing 41 (2001) 125–143

NEUROCOMPUTING

www.elsevier.com/locate/neucom

Learning efficiency improvement of back-propagation algorithm by error saturation prevention method

Hahn-Ming Lee*, Chih-Ming Chen, Tzong-Ching Huang

Department of Electronic Engineering, National Taiwan University of Science and Technology, 43 Section 4, Keelung Road, Taipei 106, Taiwan

Accepted 20 November 2000

Abstract

Back-propagation (BP) algorithm is currently the most widely used learning algorithm in artificial neural networks. With proper selection of feed-forward neural network architecture, it is capable of approximating most problems with high accuracy and generalization ability. However, the slow convergence is a serious problem when using this well-known BP learning algorithm in many applications. As a result, many researchers take effort to improve the learning efficiency of BP algorithm by various enhancements. In this research, we consider that the error saturation (ES) condition, which is caused by the use of gradient descent method, will greatly slow down the learning speed of BP algorithm. Thus, in this paper, we will analyze the causes of the ES condition in the output layer. An error saturation prevention (ESP) function is then proposed to prevent the nodes in the output layer from the ES condition. We also apply this method to the nodes in hidden layers to adjust the learning terms. By the proposed methods, we can not only improve the learning efficiency by the ES condition prevention but also maintain the semantic meaning of the energy function. Finally, some simulations are given to show the workings of our proposed method.

Keywords: Back-propagation neural network; Error saturation (ES) condition; Error saturation prevention (ESP) function; Distance entropy

1. Introduction

The back-propagation (BP) algorithm [25] is a widely used learning algorithm in artificial neural networks [11,12,14,18]. It works well for many problems (e.g.,

* Corresponding author. Tel.: + 886-2-27376307; fax: + 886-2-27376424.
E-mail address: hmlee@et.ntust.edu.tw (H.-M. Lee).

0925-2312/01/$ - see front matter
PII: S0925-2312(00)00352-0

classification or pattern recognition, etc.) [2,3,9,25]. However, it suffers two critical drawbacks from the use of gradient-descent method: one is the learning process often traps into local minimum and another is its slow learning speed. As a result, there are many researches on them [4,5,15], especially for the learning efficiency improvement by preventing the premature saturation (PS) phenomenon [16,30]. The PS means that outputs of the artificial neural networks temporarily trap into high error level during the early learning stage. In the learning issues of PS phenomenon, researchers have designed many usable modifications to solve this phenomenon [7,17,31]. However, the proposed methods are limited to many assumptions and seem to be complex (will be detailed in the next subsection). In this study, the error saturation (ES) condition [17] is considered as the main cause of PS phenomenon. The ES condition means that the nodes in each layer of artificial neural network models have outputs near the extreme value 0 or 1, but with obvious differences between the desired and actual outputs. Consequently, the learning term (signal) will be very small for the increment of weights or others parameters [23]. Therefore, we will use the error saturation prevention (ESP) method to overcome the PS phenomenon and thus the learning convergence will speed up and the local minimum problem will be relieved. Besides, we keep the semantic meaning of used mean-square-error (MSE) function to rationalize the evaluation of error criterion.

The slow learning speed of conventional BP algorithm for training feed-forward multi-layer neural network models is due to the fact that back propagation is a gradient descent method [25]. Many researchers have taken efforts on the efficiency improvement issue of BP algorithm and we will summarize them as below.

In the research of Lee et al. [16,17], they analyzed the slow learning convergence caused by the PS phenomenon. Under this situation, the learning term will be too small to lead the training weights adaptation [17,30,31]. In their presentation, they found that the PS phenomenon during early epochs of learning procedure is caused by inappropriate setting of initial weights. Also, the probability of PS can be derived by the maximum value of initial weights, the number of nodes in each layer, and the maximum slope of the sigmoid activation function. After deriving the PS probability function, they avoid the PS phenomenon by properly setting initial weights. However, they derived the PS probability equation based on the assumption that the initial weights and thresholds both are in uniform distribution. Besides, some limitations must be set during learning. In real-world applications, however, these uniform distribution requirement and limitations of learning parameters may not be easily achieved.

Vetela and Reifman [30,31] analyzed the causes of PS by dividing the learning process into three stages: beginning of saturation stage, saturation plateau stage and complete recovery stage. They proposed four necessary conditions for the occurrence of PS phenomenon by gradient of weights, the momentum terms, and extreme error condition. After constructing the PS mechanism and necessary conditions, they prevent anyone of the four conditions to be satisfied and thus avoid the slow learning condition. Actually, they think that the momentum term plays the leading role in the occurrence of PS phenomenon. Therefore, they improve the convergence speed by

properly setting momentum term in learning process. This method is efficient to prevent the PS phenomenon in the early learning stage. However, they also suffer from some limitations. For constructing the mechanism of PS phenomenon, they must design a scenario which describes the relationship between weight summation value and momentum term. The scenario assumes that the weight summation value is too small to contribute to learning process due to large quantity of momentum term in the early learning stage. By this scenario, they constructed the mechanism and necessary conditions of PS phenomenon.

Ng et al. [22], Oh [23], and Ooyen and Nienhuis [24] analyzed the causes of slow learning PS phenomenon by the gradient data of activation function. Oh [23] found that if the actual value of output nodes is under the ES condition, the learning term will be too small to improve the weights learning. By this reason, he proposed a modified energy function to overcome the drawbacks of the ES conditions. The proposed energy function will make the learning term reasonable regardless of the distribution of actual output value. Similarly, Ng et al. [22] modified the energy function to scale up the partial derivatives of the activation function and proposed a new weight evolution algorithm based on the modified energy function. Also, Ooyen and Nienhuis [24] improve the learning convergence by a new energy function based on the cross entropy (CE) algorithm. These methods could prevent the occurrence of PS phenomenon, but the used energy functions might be meaningless.

2. Error saturation prevention method

In this section, we will first describe what is the ES problem, and then analyze the causes of the ES condition and the influence related to the learning efficiency of BP algorithm. Finally, we propose an ESP function (e.g., parabolic function) to avoid the ES condition from output nodes. Then we apply the ESP method to the nodes in hidden layers with proper modifications.

2.1. Error saturation problem

In this paper, we focus on the efficiency improvement of BP algorithm by preventing the ES condition. This is because in our study we find that the main cause of the slow convergence in BP algorithm is the occurrence of the PS phenomenon, and the ES condition will result in the PS phenomenon [17,31]. The PS phenomenon is shown in Fig. 1. In this figure, we can check that during the occurrence of PS, the MSE [27] will stay high in the early learning stage (iteration 140–500) and will decrease gradually to lower error level after iteration 610. Therefore, we try to avoid the PS phenomenon by the ES prevention method to improve the learning efficiency of back propagation neural network (BPNN) models.

After studying the problem issues, we will handle the PS problem by proposing an ESP function to the learning term in nodes of output layers to overcome the error saturation phenomenon which is caused by the gradient descent method.

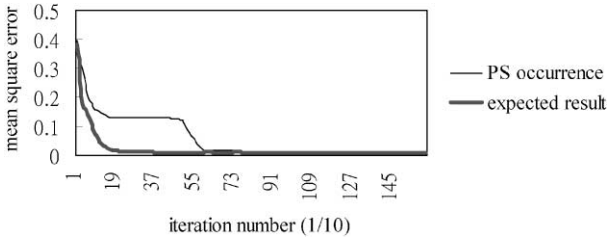

Fig. 1. The premature saturation (PS) phenomenon.

2.2. Error saturation condition in nodes of the output layer

2.2.1. Theoretic analysis

In the BP algorithm, the gradient descent method is used to search the weight solution during learning and the sigmoid activation function is used to normalize the network's output to [0,1] or to [− 1,1]. We assume that the sigmoid activation function like $f(net) = 1/(1 + e^{-k(net-\theta)})$ is used, where k means the temperature variable (cf. Oh [17] used the temperature variable T in their activation function $f(net) = 2/(1 + e^{-(1/T)(net-\theta)}) - 1$). In this paper, k is set to $1/T$ to simplify the representation, θ is the threshold of the mapped network's node, and *net* is the net input of the node.

By gradient descent method used in BP algorithm, we can get the learning term for output nodes as

$$\xi_i = (d_i - o_i)ko_i(1 - o_i), \tag{1}$$

$$\Delta w_{ji} = \eta(d_i - o_i)ko_i(1 - o_i)o_j = \eta\xi_i o_j, \tag{2}$$

where ξ_i means the learning term for the ith output node, d_i and o_i are the desired and actual values of the ith output node, respectively, Δw_{ji} is the increment of the weight from the jth node in layer $L - 1$ to the ith node in layer L, η is the learning rate, k is the temperature variable of activation function (assumed to be 1 in this paper), and o_j is the output of the jth node in layer $L - 1$. The partial structure of an ANN model is shown in Fig. 2.

In Eq. (1), we can find that when the output o_i is in extreme value close to 1 or 0, the learning term ξ_i will be very small even the delta value $(d_i - o_i)$ is large. That is, when the actual output o_i is in the extreme area, the learning term ξ_i is very small even the desired output d_i is far from the actual output o_i. Thus, we must propose some methods to avoid the ES condition.

As a result, from the distribution of the learning factor, we get an idea that if we could scale up the learning term when the actual output is near its extreme value 0 or 1, the ES condition can be prevented. Therefore, we will design a simple and effective function to speed up the learning process. Besides, we still need to keep the semantic meaning of the energy function and also keep the memory load light.

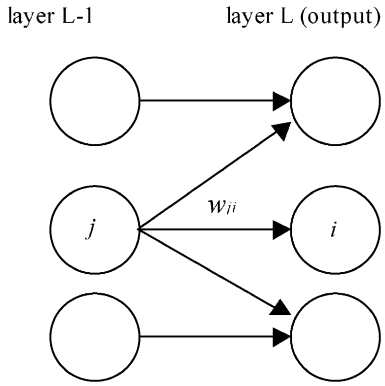

Fig. 2. The partial structure of an ANN model.

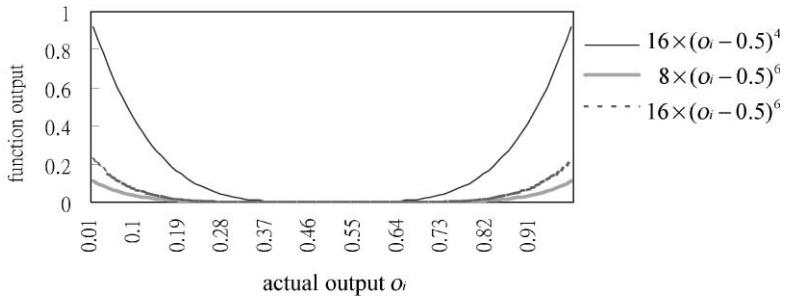

Fig. 3. Examples of the parabolic function. The parameters (α, n) are set to be (16, 4), (8, 6), and (16, 6), respectively.

2.2.2. Using the ESP function to prevent the error saturation condition

In this paper, the ESP function is given to be a parabolic function as

$$ESP(o_i) = \alpha(o_i - 0.5)^n, \tag{3}$$

where α is a scale term, n is an exponent value and o_i is the actual output of the ith node in the output layer.

The examples of parabolic function are shown in Fig. 3. In this figure, we can find that the parabolic function is indeed helpful to scale up the learning term when the actual output o_i is near the extreme value 0 or 1, and it has little influence on the learning term when the actual output is at about 0.5. Therefore, we can use the parabolic function as our ESP function. After including the ESP function, the learning term in Eq. (1) will become

$$\xi_i = (d_i - o_i)k(o_i(1 - o_i) + ESP(o_i)), \tag{4}$$

where $ESP(o_i)$ is the used parabolic function $\alpha(o_i - 0.5)^n$ as indicated in Eq. (3).

2.3. Effect of the ESP method to the energy function

In our model, for the reason to overcome the degeneration of the learning factor $o_i \times (1 - o_i)$, when the actual output o_i is in the extreme area (i.e., the occurrence of the ES condition), we speed up the learning process by including the ESP function to the learning term ξ_i. However, we should consider the effect of the ESP method to the energy function. Therefore, we will formulate the energy function when our ESP function is included. Then we analyze how the new energy function will contribute to the learning process.

2.3.1. Formulation of the energy function

In the back-propagation learning algorithm, the MSE function [25] is used as the energy function. The energy function for training pattern p is

$$E_p = \frac{1}{2} \sum_{i=1}^{N} (d_i - o_i)^2, \tag{5}$$

where N is the node number of the output layer, d_i and o_i are the desired and actual outputs for node i in the output layer. Thus,

$$E = \frac{1}{M} \sum_{p=1}^{M} E_p, \tag{6}$$

where E means the energy function (or called MSE function), and M is number of training patterns. From the above energy function and the gradient descent method, the increment of weights is shown as

$$\Delta w_{ji} = -\eta \frac{\partial E_p}{\partial w_{ji}} = -\eta (o_i - d_i) k o_i (1 - o_i) o_j, \tag{7}$$

where d_i and o_i are the desired and actual outputs of the ith output node, Δw_{ji} is the increment of the weight from the jth node in layer $L - 1$ to the ith node in layer L, η is the learning rate, k is the temperature variable of activation function (assumed to be 1 in this paper), and o_j is the output of the jth node in layer $L - 1$. The partial structure of an ANN model is shown in Fig. 2. After including the ESP function, the increment of weight will become

$$\Delta w''_{ji} = -\eta (o_i - d_i) k (o_i (1 - o_i) + ESP(o_i)) o_j. \tag{8}$$

As a result, we consider that the original energy function will be changed to

$$E''_p = E_p + E'_p, \tag{9}$$

where E_p is the MSE energy function for pattern p, E'_p is the added energy function after including our ESP function, and E''_p is the new energy function. Thus, we can rewrite the increment of weight Δw_{ji} as

$$\Delta w''_{ji} = -\eta \frac{\partial E''_p}{\partial w_{ji}} = -\eta \frac{\partial (E_p + E'_p)}{\partial w_{ji}}. \tag{10}$$

For output node i, we derive the following equations:

$$E_P = \sum_i^N E_{pi}, \tag{11}$$

$$E'_P = \sum_i^N E'_{pi}, \tag{12}$$

where E_{Pi} is the MSE energy function of output node i for pattern p, and E'_{Pi} is the added energy function of output node i after applying the ESP method for pattern p. From Eqs. (10)–(12), we could rewrite $\Delta w''_{ji}$ as follows:

$$\Delta w''_{ji} = -\eta \frac{\partial E''_p}{\partial w_{ji}} = -\eta \frac{\partial (E_p + E'_p)}{\partial w_{ji}} = -\eta \frac{\partial (E_{pi} + E'_{pi})}{\partial w_{ji}}. \tag{13}$$

Also, from Eq. (8), we can easily derive the following term:

$$\begin{aligned} \Delta w''_{ji} &= -\eta (o_i - d_i) k (o_i (1 - o_i) + ESP(o_i)) o_j \\ &= -\eta \frac{\partial (E_p)}{\partial w_{ji}} (-\eta (o_i - d_i) k ESP(o_i) o_j). \end{aligned} \tag{14}$$

From Eqs. (13) and (14), we have

$$\begin{aligned} \Delta w''_{ji} &= -\eta \frac{\partial (E_p)}{\partial w_{ji}} + (-\eta (o_i - d_i) k ESP(o_i) o_j) \\ &= -\eta \frac{\partial (E_{pi} + E'_{pi})}{\partial w_{ji}} \\ &= -\eta \frac{\partial E_{pi}}{\partial w_{ji}} + \left(-\eta \frac{\partial (E'_{pi})}{\partial w_{ji}} \right). \end{aligned} \tag{15}$$

As a result, we can have the following equation:

$$\frac{\partial E'_{pi}}{\partial w_{ji}} = (o_i - d_i) k ESP(o_i) o_j. \tag{16}$$

Thus, by the integral method, chain rule and Eq. (3), we can derive the added energy function as

$$\begin{aligned} \mathrm{E}'_{pi} = & -\alpha d_i \left[(-0.5)^n \ln o_i + \sum_{h=0}^{n-1} C_{h+1}^n \frac{1}{h+1} o_i^{h+1} (-0.5)^{n-h-1} \right] \\ & -\alpha (1 - d_i) \left[(0.5)^n \ln (1 - o_i) + \sum_{h=0}^{n-1} C_{h+1}^n \frac{1}{h+1} (o_i - 1)^{h+1} (0.5)^{n-h-1} \right] \\ & + 2 d_i (1 - d_i) - 2. \end{aligned} \tag{17}$$

From Eq. (17), energy function for patter p can be derived as

$$\begin{aligned} E''_p &= E_p + E'_p = \frac{1}{2}\sum_i^N (d_i - o_i)^2 + \sum_i^N E'_{pi} \\ &= \frac{1}{2}\sum_i^N (d_i - o_i)^2 \\ &\quad + \sum_i^N \left\{ -\alpha d_i \left[(-0.5)^n \ln o_i + \sum_{h=0}^{n-1} C_{h+1}^n \frac{1}{h+1} o_i^{h+1} (-0.5)^{n-h-1} \right] \right. \\ &\quad - \alpha(1-d_i)\left[(0.5)^n \ln(1-o_i) + \sum_{h=0}^{n-1} C_{h+1}^n \frac{1}{h+1} (o_i-1)^{h+1} (0.5)^{n-h-1} \right] \\ &\quad \left. + 2d_i(1-d_i) - 2 \right\}. \end{aligned} \tag{18}$$

Finally, the new energy function is

$$E'' = \frac{1}{M}\sum_{p=1}^M E''_p. \tag{19}$$

2.3.2. A special case of the new energy function

Without loss of generality, we use the following ESP function as an example to clarify the proposed function.

$$ESP(o_i) = 4(o_i - 0.5)^2. \tag{20}$$

As a result, the new energy function for pattern p is

$$\begin{aligned} \mathrm{E}''_p &= E_p + E'_p = \frac{1}{2}\sum_i^N (d_i - o_i)^2 \\ &\quad + \sum_{i=1}^N \{ -2(o_i - d_i)^2 - d_i \ln(o_i) - (1-d_i)\ln(1-o_i) \} \\ &= \sum_{i=1}^N \{ -1.5(o_i - d_i)^2 + d_i \ln(1/o_i) + (1-d_i)\ln(1/(1-o_i)) \}. \end{aligned} \tag{21}$$

2.3.3. Analysis of the added energy function

So far, we have derived the added energy function E'_{pi} as $-2(o_i - d_i)^2 + d_i \ln(1/o_i) + (1-d_i)\ln(1/(1-o_i))$. In this subsection, we want to see how the added energy function E'_{pi} (generated by our ESP function) will contribute to improve the learning convergence.

We can find that E'_{pi} is determined by the desired output d_i and actual output o_i. As stated before, the ES condition is caused when the actual output is near the extreme value 0 or 1 and the difference between the desired and actual output is obvious (e.g., greater than 0.8). Therefore, we can conclude that the values of the desired and actual outputs will play important roles in the ESP methods.

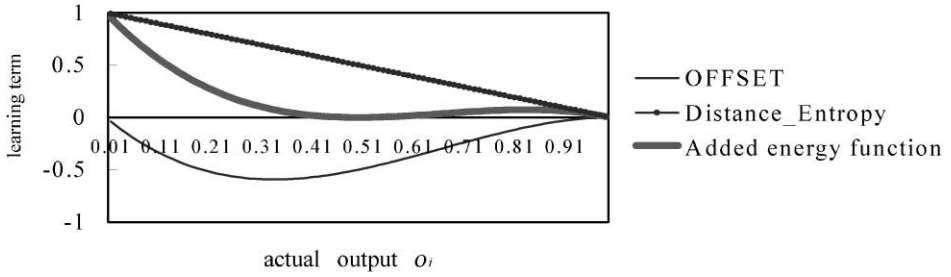

Fig. 4. The learning terms generated by the added energy function and its both terms.

From Eq. (21), we could find that the added energy function E'_{pi} consists of two terms: $-2(o_i - d_i)^2$ and $d_i \ln(1/o_i) + (1 - d_i)\ln(1/(1 - o_i))$. By the entropy theory in Shannon [26,27] and Ooyen and NienHuis [24], we can consider the term $d_i \ln(1/o_i) + (1 - d_i)\ln(1/(1 - o_i))$ as Distance_Entropy. This is because:

(1) This term is in the form of entropy function as used in Shannon [26,27] and Ooyen and NienHuis [24].
(2) $d_i \ln(1/o_i) + (1 - d_i)\ln(1/(1 - o_i))$ gives the distance measure between the desired and actual outputs.

Thus, this term will contribute to enhance the learning term when the ES condition occurs. Besides, for convenience, we name the term $-2(o_i - d_i)^2$ in Eq. (21) as OFFSET for the reason that we use it to smooth the learning term. In summary, we can reformulate the added energy function E'_{pi} as follows:

$$E'_{pi} = -2(o_i - d_i)^2 + d_i \ln(1/o_i) + (1 - d_i)\ln(1/(1 - o_i))$$
$$= \text{OFFSET} + \text{Distance\_Entropy}. \quad (22)$$

The Distance_Entropy will be $d_i \ln(1/o_i)$ when the desired output d_i is 1 or will be $(1 - d_i)\ln(1/(1 - o_i))$ when d_i is 0. In the following, we give an example whose desired output d_i is 1. As a result, the Distance_Entropy will be $d_i \ln(1/o_i)$. The Distance_Entropy will have large value as the actual output o_i is far from the desired output d_i and will have value 0 as o_i approaches d_i. That is, the Distance_Entropy enlarges the learning term when the distance between the desired and actual outputs is large. This behavior meets the characteristics of our ESP method. Thus, the Distance_Entropy reflects the learning requirement as the ES condition occurs.

To show the working of OFFSET, we give the learning terms generated by the added energy function E'_{pi} and its both terms in Fig. 4. In this figure, we can find that the learning term generated by Distance_Entropy will equal to the delta value $(d_i - o_i)$ and it is not smooth. On the other hand, by the addition of OFFSET, the learning term of added energy function would be smooth and effective to prevent the ES condition. Besides, we find the fact that the learning term of the added energy function will be zero when actual output is at about 0.5. This phenomenon meets the

characteristic of parabolic function $ESP(o_i) = 4 \times (o_i - 0.5)^2$. Thus, the learning term will be enlarged as the ES condition occurs.

From the above discussion, we could find that the added energy function E'_{pi} will not only be part of the MSE energy function, but also give the error trend of the gradient nature, i.e., the learning term will be enlarged as the ES condition occurs. As a result, we can not only improve the learning convergence but also maintain the semantic meaning of the used energy function.

2.3.4. Applying the ESP method to the learning term in the hidden layer

Without loss of generality, we assume the feed-forward neural network used is three layers, i.e., input–hidden–output layers. The learning term, which is propagated from the output layer, is derived as

$$\xi_j = \sum_i^N w_{ji}\xi_i f'(net_j) = \sum_i^N w_{ji}\xi_i\, o_j(1 - o_j), \tag{23}$$

where ξ_i and ξ_j are the learning terms of the ith output node (layer L) and the jth hidden node (layer $L-1$), w_{ji} is the weight from the jth node in layer $L-1$ to the ith node in layer L, net_j is the net value of the jth node in layer $L-1$, and o_j is the output of the jth node in layer $L-1$. The partial structure of an ANN model is shown in Fig. 2.

From Eq. (23), we can find that the learning term for the hidden layer will be small when the actual output is in the extreme area. This is the same as the ES condition in the output layer. However, there is no enough information about the desired output in hidden nodes during learning. Therefore, we cannot directly handle the ES condition as the ESP method in the output layer. From Eq. (23), we could find the following facts:

(1) The learning term for the hidden layer can be divided into two terms: one is the first derivative of activation function on the net input of the jth node in the hidden layer, i.e., the learning factor $f'(net_j) = o_j \times (1 - o_j)$, and another is the learning term propagated from nodes in the output layer $\sum_i^N w_{ji}\xi_i$.
(2) Since we have enhanced the learning term in the output layer, we will focus on the learning factor. In other words, we will overcome the degeneration caused by the learning factor to enlarge the learning term.

Therefore, for the improvement of the learning term in the hidden layer, we consider that the learning term propagated from the output layer ($\sum_i^N w_{ji}\xi_i$) is appropriate. We could also include the ESP function to the learning term in the hidden layer. In this way, we could speed up the learning convergence by the ES prevention. However, due to the unknown desired output in the hidden layer, we could not apply the ESP function directly. Since we find that the learning term in nodes of the hidden layer is too small, it may be enlarged hundreds of times if we apply the ESP function directly. This may lead the learning process to an oscillation state. Thus, we need properly control the effect of the ESP function when it is used to the learning term of the hidden layer. We will use a constant factor (a real value) to the ESP function to make the

enhancement of the learning term reasonable. Therefore, the learning term in the jth hidden node will be as

$$\xi_j = \sum_i^N w_{ji}\xi_i[o_j(1-o_j) + cESP(o_i)], \tag{24}$$

where c is a small real factor (e.g., 0.01).

2.4. Heuristics to set the parameters of the ESP function

The most important in our proposed ESP method is that we should properly determine the parameters (i.e., the scale term α and exponent value n) of the ESP function to improve the learning convergence. After analyzing the distribution of the learning term and the definition of parabolic function, we find that the exponent of the ESP function must be an even number due to the positive symmetric distribution at the input 0.5 (as shown in Fig. 3). Besides, we give the scale term α to be 2^n such that we can limit our ESP function to have real value between 0 and 1. In summary, the following strategies are suggested to determine the parameters of ESP function.

(1) As the improvement of learning convergence is not obvious, the enhancement of the ESP function is considered to be too small. In this case, the exponent value n should be decreased or the scale term α should be increased.
(2) If the learning convergence speeds up but with poor generalization ability, an oscillation phenomenon might occur in this case. Therefore, the scale term α should be decreased to reduce the effect of the ESP function. However, if this phenomenon still exists, then the exponent value n should be considered to increase.

Finally, in our following experiments, we will first apply the ESP method to the learning term in nodes of the output layer. Then we apply the ESP method to the learning term in nodes of output and hidden layers. We could find that the learning performance will speed up as we apply the ESP method to the output layer or both layers. We will detail these results in the next section.

3. Experimental results

In order to illustrate the workings of our proposed method, a modified XOR problem and two pattern classification problems with different complexity are used to verify our study in the ESP problem.

3.1. The modified XOR problem

The modified XOR problem is different from the classical XOR problem because one more pattern is included such that a unique global minimum exists [21]. Furthermore, several local minima exist simultaneously in this problem. The truth

Table 1
The truth table of the modified XOR problem

Pattern no.	Feature 1	Feature 2	Desired output
1	0.0	0.0	0.0
2	0.0	1.0	1.0
3	1.0	0.0	1.0
4	1.0	1.0	0.0
5	0.5	0.5	1.0

Table 2
Comparison results of various algorithms on the modified XOR problem

Methods	Items				
	Averaged number of iterations	Averaged training time (s)	Learning rate	Momentum term	Averaged MSE
BP algorithm	10832	17.24	0.5	—	0.0001
BP algorithm with momentum term [13]	8703	12.25	0.5	0.2	0.0001
Conjugate gradient BP [8]	7616	14.32	[a]	—	0.0001
ESP Method	4615	7.76	0.5	—	0.0001

[a]Indicates that the golden section search algorithm is used to determine a near-optimal parameter.

table of this problem is shown in Table 1. The ES problem easily occurs when the BP algorithm is used in this problem. We used a $2 \times 2 \times 1$ neural network to solve this problem. Besides, we select $\alpha = 16$ and $n = 4$ to set our ESP function, i.e., the ESP function is set to be $16(o_i - 0.5)^4$.

3.1.1. The experimental result of the ESP method in the output layer

We first apply the ESP method to the learning term in the output layer. Furthermore, our proposed ESP method is compared with other well-known enhanced BP algorithms, such as the original BP with momentum term [13], the conjugate gradient BP [7], and so on. The stop criterion is chosen as $MSE = 0.0001$. All algorithms are performed in 20 independent runs converging to the specified MSE within reasonable training epochs at initial weights given by a random process. These results are shown in Table 2. Obviously, the convergence speed of the ESP method is the fastest because the ESP mechanism is included.

3.1.2. The experimental result of the ESP method in output and hidden layers

In order to show the applicability of the ESP method in output and hidden layers, a simulation result is shown in Table 3. In this table, we compare the accelerating rates

Table 3
The accelerating rates on the modified XOR problem after applying the ESP method to output and hidden layers (compared to original BP algorithm)

Run	ESP method in the output layer	ESP method in output and hidden layers
1	2.34	2.76
2	2.78	3.15
3	2.85	3.54
Average	**2.66**	**3.15**

with the original BP after applying the ESP method to output and hidden layers on the modified XOR problem. In this case, we can find that the improvement of the learning convergence will speed up to 3.15 times. Thus, the ESP method is applicable to learning process in hidden layers.

3.2. Breast cancer data

This breast cancer data is obtained from the University of Wisconsin Hospitals [19]. We can refer the usage of this database in many researches [6,20,32,33]. There are nine attributes used to represent the pattern, and each pattern belongs to one of two classes: benign or malignant. The database consists of 367 data patterns originally [19]. Currently, it has been extended to 699 patterns. Since the original set (367 instances) is often referenced, we use the 367 patterns to be our training database. By considering the characteristics of the database, we use a $9 \times 6 \times 2$ neural network model to solve this problem. We set the node number of the hidden layer to be 6 by the heuristic [33] that the node number in the hidden layer can be equal to the average of input (9) and output (2) node numbers. In the following, we randomly divide the 367 patterns into training set (187 patterns) and testing set (180 patterns), and use $\alpha = 256$ and $n = 8$ to set our ESP function, i.e., the ESP function is set to be $256(o_i - 0.5)^8$.

3.2.1. The experimental result of the ESP method in the output layer

We first apply the ESP method to the learning term in the output layer. The error convergence diagram after applying the proposed ESP method is shown in Fig. 5. In this figure, we can see that our proposed ESP method converges rapidly to a very small MSE error at iteration 900. We can prove in this case that our ESP method will not only speed up the learning convergence, but also contribute to escape from local minimum.

All algorithms are performed in 20 independent runs converging to the specified MSE within reasonable training epochs at initial weights given by a random process. These results are shown in Table 4. Experimental results demonstrate that our proposed ESP method is superior to the original BP algorithm and the original BP algorithm with momentum term. The ESP method, however, takes more learning epochs to converge to the specified MSE than the conjugate BP algorithm does in this case. The reason is that the learning rate of the conjugate BP algorithm is dynamically

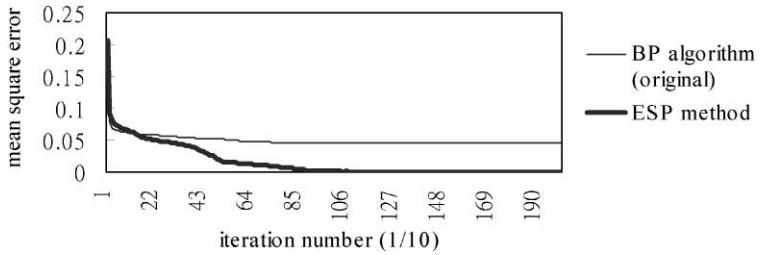

Fig. 5. The error convergence diagram after applying the proposed ESP method on breast cancer data.

Table 4
Comparison results of various algorithms on breast cancer data classification problem

Methods	Items				
	Averaged number of iterations	Averaged training time (s)	Learning rate	Momentum term	Averaged MSE
BP algorithm	4733	44.76	0.5	—	0.012
BP algorithm with momentum term [13]	3060	28.34	0.5	0.2	0.012
Conjugate gradient BP [8]	216	42.46	[a]	—	0.012
ESP Method	1460	14.81	0.5	—	0.012

[a]Indicates that the golden section search algorithm is used to determine a near-optimal parameter.

determined by a line search algorithm (the golden section search algorithm used here), but the ESP method give a constant learning rate. That is, the conjugate BP algorithm will increase the training time of each learning cycle. Therefore, although the ESP method takes more learning epochs to converge to the specified MSE than the conjugate BP algorithm does, it uses less CPU time for learning.

3.2.2. The experimental result of the ESP method in output and hidden layers

In order to show the applicability of the ESP method in output and hidden layers, a simulation result is shown in Table 5. In this table, we record the accelerating rates after applying the ESP method to output and hidden layers. In this case, we can find that the improvement of the learning convergence will speed up to 3.83 times. Thus, the ESP method is applicable to learning process in hidden layers.

3.3. Fisher iris data

In this subsection, we use Fisher iris data [10] as our third training database. Many researchers use this database to measure the performance of their proposed methods

Table 5
The accelerating rates on breast cancer data after applying the ESP method to output and hidden layers (compared to original BP algorithm)

Run	ESP method in the output layer	ESP method in output and hidden layers
1	2.97	3.20
2	3.16	3.71
3	4.30	4.57
Average	**3.48**	**3.83**

Table 6
Comparison results of various algorithms on the Fisher iris data classification problem

Methods	Items				
	Averaged number of iterations	Averaged training time (s)	Learning rate	Momentum term	MSE
BP algorithm	3386	13.95	0.9	—	0.0001
BP algorithm with momentum term [13]	2829	11.32	0.9	0.2	0.0001
Conjugate gradient BP [8]	487	12.36	[a]	—	0.0001
ESP Method	2206	9.83	0.9	—	0.0001

[a]Indicates that the golden section search algorithm is used to determine a near-optimal parameter.

[28,29]. The Fisher iris database includes 150 four-dimensional features patterns in three classes (50 for each class). We use a $4 \times 4 \times 3$ neural network model to solve the classification problem. In the following, we randomly divide the 150 patterns into training set (75 patterns) and testing set (75 patterns), and use $\alpha = 4$ and $n = 2$ to set our ESP function, i.e. the ESP function is set to be $4 \times (oi - 0.5)^2$.

3.3.1. The experimental result of the ESP method in the output layer

The simulation results of applying proposed ESP method in the output layer are shown in Table 6. Table 6 exhibits that all performed algorithms can converge to the specified MSE within reasonable training epochs at initial weights given by a random process. Similarly, experimental results demonstrate that our proposed ESP method is superior to the original BP algorithm and the original BP algorithm with momentum term. But the ESP method still takes more learning epochs to converge to the specified MSE than the conjugate BP algorithm does in this case. However, the ESP method takes less training time to converge to the specified MSE as specified before.

Table 7
The accelerating rates on Fisher iris data after applying the ESP method to output and hidden layers (compared to original BP algorithm)

Run	ESP method in the output layer	ESP method in output and hidden layers
1	1.36	1.57
2	1.47	1.89
3	1.72	2.05
Average	**1.51**	**1.84**

3.3.2. The experimental result of the ESP method in output and hidden layers

To show the applicability of the ESP method in hidden layers, a simulation result is shown in Table 7. In this table, we find that the improvement of the learning convergence will speed up to 1.84 times after the ESP method is applied simultaneously to the hidden layer and output layer.

4. Discussion

Although the proposed ESP method to improve the learning efficiency of neural network model is efficient, there are some issues needed to be investigated:

(1) *More applications by our proposed method*. In this paper, we focus on the ES prevention of classification problems whose desired output is limited (or near) to either [0,1] or [− 1,1]. However, many other problems [1] (e.g., function approximation) may suffer from the ESP condition, as well. As a result, we should extend our ESP method for other problems.

(2) *More theoretical analysis to the general ESP energy function*. In this paper, we discuss some heuristics to construct a general energy function by analyzing the new energy function generated by our ESP method. We found some important features to construct an ESP energy function. This may contribute to improve the learning convergence greatly. However, we need more knowledge to construct the general ESP energy function.

5. Conclusions

According to the previous analysis and simulations, we can draw the following advantages of applying our ESP method:

(1) The ESP method is simple and intuitive to prevent the ES condition during the learning process.
(2) The ESP method can speed up the learning efficiency in our performed problems.
(3) Distance_Entropy can explain the phenomenon of accelerating learning and the ESP method could keep the semantic meaning of the used energy function.

References

[1] S. Abe, M.S. Lan, Fuzzy rules extraction directly from numerical data for function approximation, IEEE Trans. Syst. Man Cybern. Part B: Cybernetics 25 (1) (1996) 119–129.

[2] J. Alirezaie, M.E. Jernigan, C. Nahmias, Neural network based segmentation of magnetic resonance images of the brain, 1995 IEEE Nuclear Science Symposium and Medical Imaging Conference Record, 1995, Vol. 1, pp. 1397–1401.

[3] M. Arisawa, J. Watata, Enhanced back propagation learning and it's application to business evaluation, 1994 IEEE International Conference on Neural Networks, 1994, Vol. 1, pp. 155–160.

[4] N. Baba, A new approach for finding the global minimum of error function of neural networks, Neural Networks 2 (1989) 367–373.

[5] N. Baba, A hybrid algorithm for finding global minimum of error function of neural networks, Proceeding of the International Joint Conference on Neural Networks, 1990, pp. 585–588.

[6] K.P. Bennett, O.L. Mangasarian, Robust linear programming discrimination of two linearly inseparable sets, Optimization Methods and Software 1 (1992) 23–34.

[7] K. Balakrishnan, V. Honavar, Improving convergence of back-propagation by handling flat-spots in the output layer, Proceeding of the International Conference on Artificial Neural Networks, 1992, Vol. 2, pp. 1003–1009.

[8] C. Charalambous, Conjugate gradient algorithm for efficient training of artificial neural networks, IEEE Proc. -G 139 (3) (1992) 301–310.

[9] Y.L. Cun, Back-propagation applied to handwritten code recognition, Neural Comput. 1 (1989) 541–551.

[10] R. Fisher, The Use of Multiple Measurements in Taxonomic problems, Ann. Eugenics 7 (2) (1936) 179–188.

[11] M.M. Gupta, D.H. Rao, On the principle of fuzzy neural networks, Fuzzy Sets Syst. 61 (1994) 1–18.

[12] H. Ishibuchi, R. Fujioka, H. Tanaka, Neural networks that learn from fuzzy if-then rules, IEEE Trans. Fuzzy Syst. 1 (2) (1993) 85–97.

[13] R.A. Jacobs, Increased rates of convergence through learning rate adaptation, Neural Networks 1 (1988) 295–307.

[14] A.K. Jain, J. Mao, K.M. Mohiuddin, Artificial neural networks: a tutorial, IEEE Comput. Magazine (1996) 31–44.

[15] S. Kollias, D. Anastassiou, An adaptive least squares algorithm for the efficient training of artificial neural networks, IEEE Trans. Circuits Syst. CAS-36 (1989) 1092–1101.

[16] Y. Lee, S.H. Oh, M.W. Kim, The effect of initial weights on premature saturation in back propagation learning, Proceeding of the International Joint Conference on Neural Networks, Seattle, WA, 1991, pp. 765–770.

[17] Y. Lee, S.H. Oh, M.W. Kim, An analysis of premature saturation in back-propagation learning, Neural Networks 6 (1993) 719–728.

[18] R.P. Lippmann, An Introduction to Computing with Neural Net, IEEE ASSP. Mag. (1987) 4–22.

[19] O.L. Mangasarian, W.H. Wolberg, Cancer diagnosis via linear programming, SIAM News 23 (5) (1990) 1–18.

[20] O.L. Mangasarian, R. Setiono, W.H. Wolberg, Pattern recognition via linear programming: theory and application to medical diagnosis, in: Large-Scale Numerical Optimization, T.F. Coleman, Y. Li (Eds.), SIAM Publications, Philadelphia, 1990, pp. 22–30.

[21] Marco Gori, Alberto Tesi, On the problem of local minima in backpropagation, IEEE Trans. Pattern Anal. Machine Intelligence 14(1) (1992) 76–85.

[22] S.C. Ng, S.H. Leung, A. Luk, Fast and global convergent weight evolution algorithm based on the modified back-propagation, 1995 IEEE International Conference on Neural Networks Proceedings, 1995, pp. 3004–3008.

[23] S.H. Oh, Improving the error back-propagation algorithm with a modified error function, IEEE Trans. Neural Networks 8 (3) (1997) 799–803.

[24] A.V. Ooyen, B. Nienhuis, Improving the learning convergence of the back propagation algorithm, Neural Networks 5 (1992) 465–471.

[25] D.E. Rumelhart, G.E. Hiton, R.J. Williams, Learning internal representation by error propagation, Parallel Distributed Process. 1 (1996) 318–362.
[26] C.E. Shannon, A mathematical theory of communications, Technol. J. Bell System 27 (1) (1948) 379–423.
[27] C.E. Shannon, A mathematical theory of communications, Technol. J. Bell System 27 (2) (1948) 623–656.
[28] P.K. Simpson, Fuzzy min-max neural networks, Part 1: classification, IEEE Trans. Neural Networks 3 (5) (1992) 776–786.
[29] P.K. Simpson, Fuzzy min-max neural networks Part 2: clustering, IEEE Trans. Fuzzy Systems 1 (1) (1993) 32–45.
[30] J.E. Vetela, J. Reifman, The cause of premature saturation in back propagation training, IEEE Int. Conference on Neural Networks, 1994, Vol. 2, pp. 1449–1453.
[31] J.E. Vetela, J. Reifman, Premature saturation in back-propagation networks: mechanism and necessary conditions, Neural Networks 10 (4) (1997) 721–735.
[32] W.H. Wolberg, O.L. Mangasarian, Multisurface method of pattern separation for medical diagnosis applied to breast cytology, Proceedings of the National Academy of Sciences, USA, 1990, Vol. 87, pp. 9193–9196.
[33] Yi-Cheng Ye, Applications and implementation of neural network models, Ru-Lin, Taipei, Taiwan, 1998 (in Chinese).

Hahn-Ming Lee is currently a Professor in the department of Electronic Engineering at National Taiwan University of Science and Technology, Taipei, Taiwan. He received the B.S. degree and Ph.D. degree from the department of Computer Science and Information Engineering at National Taiwan University in 1984 and 1991, respectively. His research interests includes neural networks, fuzzy computing, intelligent systems on the web, and machine learning. He is a member of IEEE, CFSA, and IICM.

Chih-Ming Chen was born in Nantou Taiwan in 1969. He received B.S. and M.S. degree from the Department of Industrial Education at National Taiwan Normal University in 1992 and 1997, respectively. Presently he is a Ph.D. candidate of the Institute of Electronic Engineering at National Taiwan University of Science and Technology. His research interests include neural networks, cerebellar model arithmetic computer, fuzzy sets theory, grey theory, and intelligent agents on the web.

Tzong-Ching Huang was born in Chia-Yi Taiwan in 1971. He received B.S. degree from the Department of Computer Science and Information Engineering at National Chao Tong University in 1994, and he received M.S. degree from the Department of Electronic Engineering at National Taiwan University of Science and Technology. Presently he is a software engineer in VIA Technology Corporation (w3.via.com.tw). His research interests include neural networks and fuzzy sets theory.