Communications
in
Mathematical
Research
26(1)(2010), 67–75

Convergence of Online Gradient Method with Penalty for BP Neural Networks*

Shao Hong-mei[1], Wu Wei[2] and Liu Li-jun[3]

(*1. College of Mathematics and Computational Science, China University of Petroleum, Dongying, Shandong, 257061*)

(*2. Department of Applied Mathematics, Dalian University of Technology, Dalian, Liaoning, 116024*)

(*3. Department of Mathematics, Dalian Nationalities University, Dalian, Liaoning, 116605*)

Communicated by Ma Fu-ming

Abstract: Online gradient method has been widely used as a learning algorithm for training feedforward neural networks. Penalty is often introduced into the training procedure to improve the generalization performance and to decrease the magnitude of network weights. In this paper, some weight boundedness and deterministic convergence theorems are proved for the online gradient method with penalty for BP neural network with a hidden layer, assuming that the training samples are supplied with the network in a fixed order within each epoch. The monotonicity of the error function with penalty is also guaranteed in the training iteration. Simulation results for a 3-bits parity problem are presented to support our theoretical results.

Key words: convergence, online gradient method, penalty, monotonicity

2000 MR subject classification: 92B20, 68T05

Document code: A

Article ID: 1674-5647(2010)01-0067-09

1 Introduction

Online gradient method (OGM for short) is a popular and commonly used learning algorithm for training the weights of BP networks (see [1]–[5]). Penalty methods are often introduced into the training procedure and have proved efficient to improve the generalization performance and to decrease the complexity of neural networks (see [6]–[12]). Here the generalization performance refers to the capacity of a neural network to give correct outputs for untrained data. A simple and commonly used penalty added to the conventional error function is the squared penalty, a term proportional to the magnitude of the network weights

*Received date: Dec. 14, 2007.

Foundation item: The NSF (10871220) of China and the Doctoral Foundation (Y080820) of China University of Petroleum.

(see [2] and [3]). Applied to the weight updating rule of batch gradient descent algorithm, the influence of penalty on the training can be seen clearly:

$$\Delta w(n) = -\eta \frac{\partial E(w)}{\partial w(n)} - \lambda w(n) \tag{1.1}$$

where w, $\Delta w(n)$, $E(w)$, η and λ represent the vector of all weights, the modification of w at the n-th iteration, the conventional error function, the learning rate and the penalty parameter, respectively. As shown in (1.1), in addition to the update by the gradient algorithm, the weight is decreased by λ times of its old value. Consequently, the weights with small magnitudes are encouraged to decrease to zero and those with large magnitudes are constrained from growing too large during the training process. This will force the network response to be smoother and less likely to overfit, leading to good generalization (see [6], [7] and [11]). Many experiments have shown that as well as being beneficial from a generalization capacity prospective, such a term provides a way to control the magnitude of the weights during the training procedure in literature (see [6] and [10]–[12]). But there remains a lack of theoretical assurance on this experimental observation, especially for online cases.

For simplicity of analysis, the input sample ξ^j is provided to the network in a fixed order in each training epoch. We shall show that the online gradient method with penalty and fixed inputs (OGM-PF) is deterministically convergent. A boundedness theorem is established for the network weights connecting the input and hidden layers, which is also a desired outcome of adding penalty. Another key point of our proofs lies in the monotonicity of the error function with such a penalty term during the training iteration.

In this paper, $\|\cdot\|$ stands for the Euclidean norm and C_i stand for suitable positive constants which are independent of the iteration step n.

The rest of this paper is organized as follows. OGM-PF is described in detail in Section 2. The main theorems are presented in Section 3. In Section 4, the algorithm OGM-PF is applied to a 3-bits parity problem to illustrate our theoretical findings. Some lemmas and detailed proofs of the theorems are gathered as an Appendix.

2 Online Gradient Method with Penalty

Consider a three-layer BP network consisting of p input units, q hidden units and one output unit. Let $w_0 = (w_{01}, \cdots, w_{0q})$ be the weights between the hidden units and the output unit, and $w_i = (w_{i1}, \cdots, w_{ip})$ be the weights between the input units and the hidden unit i ($i = 1, 2, \cdots, q$). To simplify the presentation, we write all the weight parameters in a compact form, i.e., $W = (w_0, w_1, \cdots, w_q) \in \mathbf{R}^{q+pq}$. And we define a matrix $V = (w_1^T, \cdots, w_q^T)^T \in \mathbf{R}^{q\times p}$ and a vector function $G(x) = (g(x_1), \cdots, g(x_q))$ for $x = (x_1, \cdots, x_q) \in \mathbf{R}^q$. Assume that $\{\xi^j, O^j\}_{j=1}^J$ is the given set of training samples and $g : \mathbf{R} \to \mathbf{R}$ is a transfer function for both hidden and output layers. Then for each input $\xi \in \mathbf{R}^p$, the actual output vector of the hidden layer is $G(V\xi)$ and the final output of the network is $\zeta = g(w_0 \cdot G(V\xi))$. A

conventional square error function is given by

$$\tilde{E}(W) = \frac{1}{2}\sum_{j=1}^{J}\big(O^j - g(w_0 \cdot G(V\xi^j))\big)^2. \tag{2.1}$$

By adding a penalty term, the total error function takes the form (see [3])

$$\begin{aligned} E(W) &= \tilde{E}(W) + \lambda\sum_{i=0}^{q}\|w_i\|^2 \\ &\equiv \sum_{j=1}^{J} g_j\big(w_0 \cdot G(V\xi^j)\big) + \lambda\sum_{i=0}^{q}\|w_i\|^2. \end{aligned} \tag{2.2}$$

Differentiating $E(W)$ with respect to W gives

$$E_W(W) = \big(E_{w_0}(W), E_{w_1}(W), \cdots, E_{w_q}(W)\big)^T, \tag{2.3}$$

where

$$E_{w_0}(W) = \sum_{j=1}^{J} g_j'\big(w_0 \cdot G(V\xi^j)\big)G(V\xi^j) + 2\lambda w_0,$$

$$E_{w_i}(W) = \sum_{j=1}^{J} g_j'\big(w_0 \cdot G(V\xi^j)\big)w_{0i}g'(w_i \cdot \xi^j)\xi^j + 2\lambda w_i, \qquad i = 1, 2, \cdots, q.$$

The online gradient algorithm updates the weights after the presentation of each training sample ξ^j. As done in [13], we can choose the training samples in a fixed order and the online gradient method with penalty (OGM-PF) can be described as follows:

$$W^{nJ+j} = W^{nJ+j-1} + \Delta_j^n W^{nJ+j-1}, \tag{2.4}$$

where

$$\Delta_j^n w_0 = -\eta_n\Big[g_j'(w_0 \cdot G(V\xi^j))G(V\xi^j) + \frac{2\lambda}{J}w_0\Big],$$

$$\Delta_j^n w_i = -\eta_n\Big[g_j'(w_0 \cdot G(V\xi^j))w_{0i}g'(w_i \cdot \xi^j)\xi^j + \frac{2\lambda}{J}w_i\Big].$$

Here η_n is the learning rate in the n-th training epoch. From an initial value $\eta_0 \in (0, 1]$, it changes its value after each epoch of training according to (see [13] and [14])

$$\frac{1}{\eta_n} = \frac{1}{\eta_{n-1}} + \beta, \qquad n = 1, 2, \cdots, \tag{2.5}$$

where $\beta > 0$ is a constant to be specified in assumption (A3) below.

3 Main Theorems

The following assumptions are needed.

(A1) $|g(t)|$, $|g'(t)|$ and $|g''(t)|$ are uniformly bounded for $t \in \mathbf{R}$.

(A2) $\|w_0^k\|$ $(k = 0, 1, \cdots)$ are uniformly bounded.

(A3) Inequality (5.12) is valid, and η_0 and β satisfy $0 < \eta_0 < \dfrac{1}{\beta_0}$ and $\beta_0 \le \beta < \dfrac{1}{\eta_0}$, where β_0 is a previously chosen constant.

(A4) There exists a closed bounded region $\Phi \subset \mathbf{R}^{q+pq}$ such that all the weights $\{W^k\} \subset \Phi$, and the set

$$\Phi_0 = \{W \in \Phi : E_W(W) = 0\}$$

contains only finite points.

Remark 3.1 We note that from (2.2) and (A1), the functions $|g_j(t)|$, $|g_j'(t)|$ and $|g_j''(t)|$ are also uniformly bounded for all t and j.

Theorem 3.1 (Monotonicity) *Let the error function $E(W)$ be given in* (2.2)*, the learning rates $\{\eta_n\}$ be determined by* (2.5)*, W^0 be an arbitrary initial value, and the weights $\{W^k\}$ be generated by the algorithm OGM-PF* (2.4)*. If assumptions* (A1)–(A3) *are valid, then*

$$E\big(W^{(n+1)J}\big) \le E\big(W^{nJ}\big), \qquad n = 0, 1, \cdots. \tag{3.1}$$

Theorem 3.2 (Boundedness) *Under the same assumptions of Theorem* 3.1*, the weight sequences $\{w_i^k\}_{k=0}^{\infty}$ $(i = 1, 2, \cdots, q)$ connecting the input and hidden layers is uniformly bounded.*

Theorem 3.3 (Convergence) *Let the error function $E(W)$ be defined in* (2.2) *and the weights $\{W^k\}$ be updated by the algorithm OGM-F* (2.4)*. If assumptions* (A1)–(A3) *are satisfied, then there holds the following weak convergence result*:

$$\lim_{k\to\infty} \|E_W(W^k)\| = 0. \tag{3.2}$$

Furthermore, if assumption (A4) *is also valid, we have the strong convergence*: *There exists $W^* \in \Phi_0$ such that*

$$\lim_{k\to\infty} W^k = W^*. \tag{3.3}$$

4 Numerical Experiment

To demonstrate the convergence behavior of the online gradient method with penalty used in this paper, a benchmark problem−parity problem is simulated. The parity problem is a well-known difficult problem that has often been used for testing the performance of network training algorithm.

The input set consists of 2^n patterns in n-dimensional space and each pattern is an n-bit binary vector. The target value O^j is equal to 1 if the number of 1 in the pattern is odd, otherwise it is equal to 0. For simplicity, in this experiment we use the 3-bit parity problem which can be solved by a three-layer network with the structure 3-3-1. The transfer function for both the hidden layer and output layer is chosen to be logsig$(\cdot)$. This test is carried out by setting the initial learning rate η_0 and the penalty parameter λ with different values, varying from 0.9 to 0.5, and 0.01 to 0.001, respectively. For every combination of different η_0 and λ, the training starts with the same initial weights.

Since the changes of total error (see (2.2)) and network weights become quite tiny when the number of iteration exceeds 200, and their convergence performances are similar, we just lay out the observation with $\eta_0 = 0.8$ in the first 200 epoches.

As shown in Fig. 4.1, the total error with penalty decreases monotonically and the corresponding gradient tends to zero as the number of iteration increases. The restraint of the penalty term on the magnitude of the weights is shown in Fig. 4.2. Without penalty, the weight becomes larger and larger during the training iteration, while the magnitude of the weights is effectively reduced and finally tends to keep steady after adding the penalty.

Table 4.1 summarizes the results obtained at the 200-th iteration by taking different penalty parameters, from which we see that the larger λ is, the smaller the weight becomes. Hence, the penalty approach provides a mechanism to effectively control the magnitude of the weights.

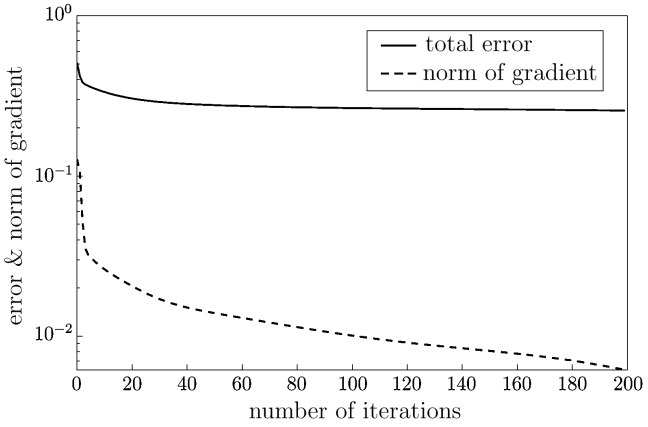

Fig. 4.1 Total error and norm of gradient with penalty

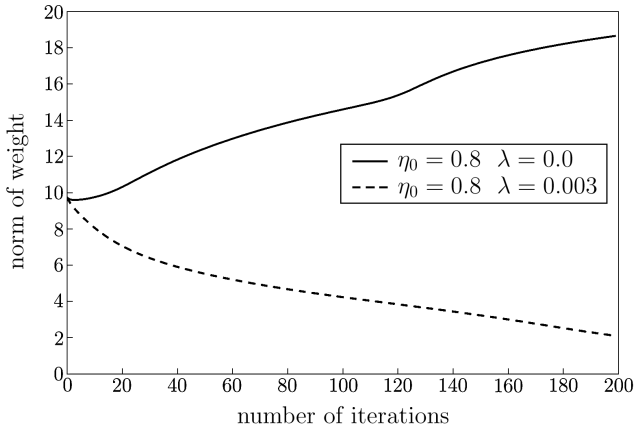

Fig. 4.2 Effect of λ on error and weight

Table 4.1 Effect of λ on error and weight

$\eta_0 = 0.8$	Square error	Total error	$\|W\|$
$\lambda = 0$	0.005816	0.00581	18.33437
$\lambda = 0.001$	0.03360	0.12896	13.81066
$\lambda = 0.003$	0.22003	0.26617	5.54667
$\lambda = 0.005$	0.24999	0.25235	0.97141
$\lambda = 0.006$	0.24999	0.25228	0.87349
$\lambda = 0.007$	0.25000	0.25220	0.79435
$\lambda = 0.008$	0.25000	0.25212	0.72878
$\lambda = 0.009$	0.250009	0.25204	0.67341
$\lambda = 0.01$	0.25000	0.25195	0.62597

5 Appendix

We introduce the following notations to simplify arguments:

$$r_i^{n,j} = \Delta_j^n w_i^{nJ+j-1} - \Delta_j^n w_i^{nJ}, \tag{5.1}$$

$$G^{n,j} = G(V^n \xi^j), \qquad \psi^{n,l,j} = G^{nJ+l-1,j} - G^{nJ,j}, \tag{5.2}$$

$$d_i^{n,j} = w_i^{nJ+j-1} - w_i^{nJ}, \qquad D^{n,j} = W^{nJ+j-1} - W^{nJ}. \tag{5.3}$$

Lemma 5.1 *Let $\{\eta_n\}$ be given by (2.5). Then the following estimates hold:*

$$0 < \eta_n < \eta_{n-1} \le 1, \qquad n = 1, 2, \cdots, \tag{5.4}$$

$$\eta_n < \frac{\rho}{n}, \quad \rho = \frac{1}{\beta}, \qquad n = 1, 2, \cdots, \tag{5.5}$$

$$\eta_n > \frac{\tau}{n}, \quad \tau = \frac{\eta_0}{1 + \eta_0 \beta}, \qquad n = 1, 2, \cdots. \tag{5.6}$$

Proof. This lemma can be proved by using (2.5) and $\eta_0 \in (0, 1]$.

Next, we present a few more lemmas. Their proofs are omitted to save the space. Proofs for similar results can be found in [13] and [14].

Lemma 5.2 *Let assumptions* (A1) *and* (A2) *be valid. There are $C_i > 0$ such that*

$$\|G(x)\| \le C_1, \qquad x \in \mathbf{R}^q, \tag{5.7}$$

$$\|\psi^{n,l,j}\| \le C_2 \Big(\sum_{i=1}^{q} \Big\| \sum_{k=1}^{l-1} \Delta_k^n w_i^{nJ} \Big\| + \sum_{i=1}^{q} \sum_{k=1}^{l-1} \|r_i^{n,k}\| \Big), \tag{5.8}$$

$$\sum_{i=0}^{q} \sum_{j=1}^{J} \|r_i^{n,j}\| \le C_3 \eta_n \sum_{i=0}^{q} \sum_{j=1}^{J} \|\Delta_j^n w_i^{nJ}\|. \tag{5.9}$$

Lemma 5.3 *There exists a positive constant γ independent of n such that*

$$E\big(W^{(n+1)J}\big) \le E\big(W^{nJ}\big) - \frac{1}{\eta_n} \sum_{i=0}^{q} \Big\| \sum_{j=1}^{J} \Delta_j^n w_i^{nJ} \Big\|^2 + \gamma \sum_{i=0}^{q} \sum_{j=1}^{J} \big\| \Delta_j^n w_i^{nJ} \big\|^2. \tag{5.10}$$

In virtue of (5.10), Theorem 3.1 can be proved if for any nonnegative integer n there holds

$$\frac{1}{\eta_n} \sum_{i=0}^{q} \Big\| \sum_{j=1}^{J} \Delta_j^n w_i^{nJ} \Big\|^2 \ge \gamma \sum_{i=0}^{q} \sum_{j=1}^{J} \big\| \Delta_j^n w_i^{nJ} \big\|^2. \tag{5.11}$$

For $n = 0$, if the left side of (5.11) is zero, then

$$\|E_W(W^0)\| = 0.$$

Hence, we have already reached a local minimum of the error function, and the iteration can be terminated. Otherwise,

$$\frac{1}{\eta_0} \sum_{i=0}^{q} \Big\| \sum_{j=1}^{J} \Delta_j^0 w_i^0 \Big\|^2 \ge \gamma \sum_{i=0}^{q} \sum_{j=1}^{J} \big\| \Delta_j^0 w_i^0 \big\|^2 \tag{5.12}$$

will be satisfied for all small enough η_0. In this case, we can prove (5.11) by applying an induction on n, resulting in the next lemma.

Lemma 5.4 *Let assumptions* (A1) *and* (A2) *be satisfied and* $\{\eta_n\}$ *be updated by* (2.5). *Then there exists a constant* β_0 *such that if the initial learning rate* η_0 *and the constant* β *in* (2.5) *satisfy Assumption* (A3), *then* (5.11) *holds for all* n.

The next two lemmas are necessary to our convergence result. Their proofs are omitted since they are quite similar to those of Lemma 3.5 in [14] and Theorem 14.1.5 in [15] (also in [16]), respectively.

Lemma 5.5 *Suppose that the series* $\sum_{n=1}^{\infty}\frac{a_n^2}{n}<\infty$, *that* $a_n>0$ *for* $n=1,2,\cdots$, *and that there exists a constant* $\mu>0$ *such that* $|a_{n+1}-a_n|<\frac{\mu}{n}$, $n=1,2,\cdots$. *Then* $\lim_{n\to\infty}a_n=0$.

Lemma 5.6 *Let* $F:\Omega\subset\mathbf{R}^n\to\mathbf{R}^m$ $(n,m\geq 1)$ *be continuous* ($\Omega\subset\mathbf{R}^n$ *is a closed bounded region*) *and* $\Omega_0=\{x\in\Omega:F(x)=0\}$ *be finite. Suppose that the sequence* $\{x_k\}\subset\Omega$ *is such that*

(1) $\lim_{k\to\infty}F(x_k)=0$;

(2) $\lim_{k\to\infty}\|x_{k+1}-x_k\|=0$.

Then, there exists $x_*\in\Omega_0$ *such that* $\lim_{k\to\infty}x_k=x_*$.

Now we are ready to prove Theorems 3.1–3.3.

Proof of Theorem 3.1 The monotonicity theorem can be drawn directly from (5.12), Lemma 5.3 and Lemma 5.4.

Proof of Theorem 3.2 In light of (2.2) and (3.1), there holds for all $n=0,1,\cdots$ and $i=1,2,\cdots,q$ that

$$\left\|w_i^{nJ}\right\|\leq\sqrt{\frac{1}{\lambda}E(W^0)}\equiv M_0. \tag{5.13}$$

A combination of (A1), (A2), (5.4) and (5.13) gives

$$\left\|w_i^{nJ+1}\right\|\leq M_0+\left(C_4+\frac{2\lambda}{J}M_0\right)\equiv M_1, \tag{5.14}$$

where

$$C_4=\sup_{t\in\mathbf{R},1\leq j\leq J}|g_j'(t)|\sup_{t\in\mathbf{R}}|g'(t)|\sup_{k\geq 0}\|w_0^k\|\max_{1\leq j\leq J}\|\xi^j\|.$$

Similarly, we have

$$\left\|w_i^{nJ+2}\right\|\leq M_1+\left(C_4+\frac{2\lambda}{J}M_1\right)\equiv M_2 \tag{5.15}$$

and there are positive integers M_j $(3\leq j\leq J)$ such that

$$\left\|w_i^{nJ+j}\right\|\leq M_j,\qquad n=0,1,\cdots;\ i=1,2,\cdots,q. \tag{5.16}$$

On setting $M=\max\{M_0,M_1,\cdots,M_J\}$, (5.13)–(5.16) state that for all n, i and j

$$\left\|w_i^{nJ+j}\right\|\leq M. \tag{5.17}$$

The upper bound M is obviously independent of n, i and j, which indicates the boundedness of $\{w_i^k\}_{k=0}^{\infty}$ $(i=1,\cdots,q)$ and $\{V^k\}_{k=0}^{\infty}$.

Proof of Theorem 3.3 Set

$$\sigma^n=\frac{1}{\eta_n}\sum_{i=0}^{q}\|\sum_{j=1}^{J}\Delta_j^n w_i^{nJ}\|^2-\gamma\sum_{i=0}^{q}\sum_{j=1}^{J}\|\Delta_j^n w_i^{nJ}\|^2.$$

It follows from (5.12) and Lemma 5.4 that $\sigma_n \geq 0$ for all $n = 0, 1, \cdots$. By virtue of (5.10) we can write

$$E\big(W^{(n+1)J}\big) \leq E\big(W^{J}\big) - \sum_{k=1}^{n} \sigma^k. \tag{5.18}$$

Letting $n \to \infty$ gives

$$\sum_{n=1}^{\infty} \sigma^n \leq E\big(W^{J}\big) < \infty. \tag{5.19}$$

Use assumptions (A1), (A2), (5.5) and Theorem 3.2 to show

$$\sum_{n=1}^{\infty} \Big(\gamma \sum_{i=0}^{q} \sum_{j=1}^{J} \big\|\Delta_j^n w_i^{nJ}\big\|^2\Big) < \rho^2 C_5 \sum_{n=1}^{\infty} \frac{1}{n^2} < \infty. \tag{5.20}$$

This together with (5.19) and (5.6) leads to

$$\begin{aligned}\sum_{n=1}^{\infty} \frac{1}{n}\big\|E_W(W^{nJ})\big\|^2 &< \frac{1}{\tau} \sum_{n=1}^{\infty} \Big(\frac{1}{\eta_n} \sum_{i=0}^{q} \big\| \sum_{j=1}^{J} \Delta_j^n w_i^{nJ}\big\|^2\Big) \\ &< \infty. \end{aligned} \tag{5.21}$$

In light of (A1), (A2), (2.4), (5.5) and Theorem 3.2, there is $C_6 > 0$ such that

$$\sum_{i=0}^{q} \big\|d_i^{n,J+1}\big\| \leq \sum_{i=0}^{q} \sum_{j=1}^{J} \big\|\Delta_j^n w_i^{nJ+j-1}\big\| < \frac{C_6}{n}. \tag{5.22}$$

Applying the mean value theorem to each $g_j'(t)$ and $g'(t)$, we obtain from assumptions (A1), (A2) and Inequality (5.22) that

$$\Big|\big\|E_W(W^{(n+1)J})\big\| - \big\|E_W(W^{nJ})\big\|\Big| < \frac{C_7}{n}. \tag{5.23}$$

A combination of (5.21), (5.23) and Lemma 5.5 leads to

$$\lim_{n\to\infty} \big\|E_W(W^{nJ})\big\| = 0. \tag{5.24}$$

Similarly, as (5.23), there is $C_8 > 0$ such that for all $j = 1, 2, \cdots, J-1$,

$$\big\|E_W(W^{nJ+j}) - E_W(W^{nJ})\big\| < \frac{C_8}{n}. \tag{5.25}$$

Accordingly, we can state that

$$\lim_{n\to\infty} \big\|E_W(W^{nJ+j})\big\| = 0, \qquad j = 1, 2, \cdots, J-1. \tag{5.26}$$

Let us put (5.24) and (5.26) together and express them in a compact form, i.e.,

$$\lim_{k\to\infty} \big\|E_W(W^{k})\big\| = 0. \tag{5.27}$$

The proof of the strong convergence can be done as that of Theorem 3 in [17], and the detail is omitted.

References

[1] Rumelhart, D. E., McClelland, J. L. and the PDP Research Group, Parall Distributed Processing-Explorations in the Microstructure of Cognition, Mass., MIT Press, Cambridge, 1986.

[2] Haykin, S., Neural Networks: A Comprehensive Foundation, Macmillan, New York, 1994.

[3] Hinton, G. E., Connectionist learning procedures, *Artif. Intell.*, **40**(1989), 185–234.

[4] Sollich, P. and Barber, D., Online learning from finite training sets and robustness to input bias, *Neural Comput.*, **10**(1998), 2201–2217.

[5] Luo, Z., On the convergence of the LMS algorithm with adaptive learning rate for linear feedforward networks, *Oper. Res.*, **3**(1991), 226–295.

[6] Setiono, R., A penalty-function approach for pruning feedforward neural networks, *Neural Comput.*, **9**(1997), 185–204.

[7] Reed, R., Pruning algorithms–a survey, *IEEE Trans. Neural Networks*, **4**(1993), 740–747.

[8] Chen, Z. and Haykin, S., On different facets of regularization theory, *Neural Comput.*, **14**(2002), 2791–2846.

[9] Ishikawa, M., Structural learning with forgetting, *Neural Networks*, **9**(1996), 509–521.

[10] Cho, S. and Chow, T. W., Training multilayer neural networks using fast global learning algorithm–least squares and penalized optimization methods, *Neurocomputing*, **25**(1999), 115–131.

[11] Takase, H., Kita, H. and Hayashi, T., Effect of regularization term upon fault tolerant training, *Proc. Int. Joint Conf. Neural Networks*, **2**(2003), 1048–1053.

[12] Saito, K. and Nakano, R., Second-order learning algorithm with squared penalty term, *Neural comput.*, **12**(2000), 709–729.

[13] Wu, W., Feng, G., Li, Z. and Xu, Y., Convergence of an online gradient method for BP neural networks, *IEEE Trans. Neural Networks*, **16**(2005), 533–540.

[14] Wu, W., Feng, G. and Li, X., Training multilayer perceptrons via minimization of sum of ridge functions, *Adv. Comput. Math.*, **17**(2002), 331–347.

[15] Ortega, J. M. and Rheinboldt, W. C., Iterative Solution of Nonlinear Equations in Several Variables, Academic Press, New York, 1970.

[16] Yuan, Y. and Sun, W., Optimization Theory and Methods, Science Press, Beijing, 2001.

[17] Wu, W., Shao, H. and Qu, D., Strong Convergence for Gradient Methods for BP Networks Training, Proc. Internat. Conf. Neural Network Brains (ICNNB'05), IEEE Press, 2005, 332–334.