DE GRUYTER
OPEN

JACSM 2016, Vol. 8, No. 2, pp. 85 - 98

10.1515/jacsm-2016-0006

CONVERGENCE ANALYSIS OF INVERSE ITERATIVE NEURAL NETWORKS WITH L_2 PENALTY

Yanqing Wen[1], Jian Wang[1], Bingjia Huang[1] and Jacek M. Zurada[2,3]

[1]College of Science, China University of Petroleum, Qingdao 266580, China
wangjiannl@upc.edu.cn; hbjia@upc.edu.cn

[2]Department of Electrical and Computer Engineering, University of Louisville,
Louisville, KY, 40292, USA)

[3]Information Technology Institute, University of Social Sciences,
Łodz 90-113, Poland
jacek.zurada@louisville.edu

Abstract

The iterative inversion of neural networks has been used in solving problems of adaptive control due to its good performance of information processing. In this paper an iterative inversion neural network with L_2 penalty term has been presented trained by using the classical gradient descent method. We mainly focus on the theoretical analysis of this proposed algorithm such as monotonicity of error function, boundedness of input sequences and weak (strong) convergence behavior. For bounded property of inputs, we rigorously proved that the feasible solutions of input are restricted in a measurable field. The weak convergence means that the gradient of error function with respect to input tends to zero as the iterations go to infinity while the strong convergence stands for the iterative sequence of input vectors convergence to a fixed optimal point.

Keywords: neural networks; gradient descent; inverse iterative; monotonicity; regularization; convergence

1 Introduction

Artificial neural networks have been widely used in cognitive science, computational intelligence and intelligent information processing [1, 2]. Feedforward neural networks are some of the most popular networks whose learning modes and theoretical properties are studied in numerous reports [3-5]. Backpropagation (BP) algorithm is the most broadly applied technique to train the feedforward neural networks. For BP networks, there are one or more hidden layers, in which the adjacent layer are fully connected with weights. Gradient descent methods are often employed to find the optimal solutions by charging weights in the descent direction of objective function. Generally speaking, there are three main drawbacks of this classical BP networks: slow convergence, poor generalization and local optimal solution.To overcome these obstacles, many training improvements for BP networks have been suggested such as adding penalty terms (regularization), adaptive adjustment of learning rate and introducing momentum terms [6-10]. Actually, it is a common strategy to improve the generalization and prune more redundant weights through regularization method.

Inverse problem is one of the most important mathematical problems which tells us about parameters that cannot be directly observed [11, 12]. It is the inverse of a forward problem which deals with the results and then compute the input. Contrary to the feedforword neural networks which correspond to the forward problem, the inverse problem results in iterative inversion of neural networks.

For BP algorithm, the output error is propagated backward through the network and the error is computed by the weights. Conversely, an iterative inversion algorithm has been proposed in [13], where the weights learning is replaced by inputs learning. In this approach, errors in the network output are described with the network inputs. In addition, this iterative inversion algorithm trains by the gradient descent method. In order to solve the optimization problem of electromagnetic mechanism, a novel inverse network has been designed which effectively avoids the local minimum problem [14]. Similar to the Bonhoeffer-Van der Pol (BVP) model, an inverse function delayed network is presented by the use of anti-delay function model. It demonstrates that this proposed network can quickly converge to the optimal solution of combinatorial optimization problems. In [18], a real-time inversion of neural network has been described by combining the particle swarm method. The reconfigurable implementation of network inversion effectively reduced the computation time to near real-time levels.

For trained neural networks, over-fitting is a common problem which leads to poor generalization. To overcome this problem, a typical technique is to employ the regularization method, that is, introduce the penalty term [12-17]. We note that L_2 norm of the parameters is one of the most often used penalty terms. There are many researches on L_2 regularization which demonstrate the it can produce smooth solution and effectively control the magnitude of the parameters [14, 15, 16, 18].

As we know, the iterative inversion of neural networks has been widely used in real applications. However, it is necessary to pay attention to its theoretical analysis. In [19], an iterative inversion algorithm of neural networks with momentum has been designed and its convergence results are proved in detail. However, the boundedness of inputs can not be guaranteed which may lead to a very large solution.

In this paper, we focus on the iterative inversion algorithm of neural networks with L_2 penalty term. The monotonicity of error function has been proved which shows that the objective functions of input are decreasing along with the iterations. More importantly, the boundedness of the inputs are rigorously proved through introducing the L_2 penalty term. Furthermore, both the week and strong convergence results are obtained, that is, the gradient of error function with respect to input vector approaches zero and the iterative input sequence converges to a fixed optimal point as the iterations go to infinity.

The rest of the paper is organized as follows: in Section 2, the iterative version algorithm with L_2 penalty is presented. In Section 3, the proofs of the theoretical results are demonstrated in detail. Finally, we conclude the paper with some useful remarks in Section 4.

2 Inverse iterative algorithms with L_2 penalty

Let us begin with an introduction of an inverse iterative algorithms for neural network with three layers. The numbers of neurons for the input, hidden and output layers are p, n and 1, respectively, suppose that the input sample and the corresponding ideal output sample are $\mathbf{x} \in \mathbf{R}^p$ and $O \in \mathbf{R}$.

Let $\mathbf{V} = \left(v_{ij}\right)_{n \times p}$ be the weight matrix connecting the input and the hidden layers, and write $\mathbf{v}_i = \left(v_{i1}, \cdots, v_{ip}\right)^T \in \mathbf{R}^p$ $i = 1, 2, \cdots, n$. The weight vector connecting the hidden and the output layers is denoted by $\mathbf{w} = (w_1, w_2, \cdots, w_n)^T \in \mathbf{R}^n$. Let $g : \mathbf{R} \to \mathbf{R}$ be given activation functions for the hidden and output layers. For convenience, we introduce the following vector valued function

$$G\left(\mathbf{u}\right) = \left(g(u_1), g(u_2), \cdots, g(u_n)\right)^T \tag{1}$$

For any given input $\mathbf{x} \in R^p$, the output of the hidden neurons is $G(\mathbf{V}\mathbf{x})$, and the final actual output is

$$y = g\left(\mathbf{w} \cdot G(\mathbf{V}\mathbf{x})\right) \tag{2}$$

The error function with L_2 regularization penalty term is

$$E\left(\mathbf{x}\right) = \frac{1}{2}\left(O - g\left(\mathbf{w} \cdot G\left(\mathbf{V}\mathbf{x}\right)\right)\right)^2 + \lambda \left\|\mathbf{x}\right\|_2^2 \tag{3}$$

where $\|\mathbf{x}\|_2^2 = \sum_{j=1}^{p} |x_j|^2$.

The purpose of inverse iterative algorithms is for the given output $O \in \mathbf{R}$, input $\mathbf{x}$ makes error function $E(\mathbf{x})$ to achieve its minimal value. To simplify the writing, we do the following transformation

$$\tilde{g}(t) = \frac{1}{2}(O - g(t))^2, \quad t \in R \tag{4}$$

The gradient of the error function with respect to $\mathbf{x}$ is given by

$$E_{\mathbf{x}}(\mathbf{x}) = \tilde{g}'(\mathbf{w} \cdot G(\mathbf{Vx})) \sum_{i=1}^{n} w_i g'(\mathbf{v}_i \cdot \mathbf{x}) \mathbf{v}_i + \lambda \nabla\left(\|\mathbf{x}\|_2^2\right) \tag{5}$$

where $\nabla\left(\|\mathbf{x}\|_2^2\right) = (2x_1, 2x_2, \cdots, 2x_p)^T$.

Given an initial input vector $\mathbf{x}^0 \in R^p$, inverse iterative algorithms with L_2 penalty updates the inputs iteratively by the formula

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \eta E_{\mathbf{x}}(\mathbf{x}^k) \tag{6}$$

$$= \mathbf{x}^k - \eta[\tilde{g}'(\mathbf{w} \cdot G(\mathbf{Vx})) \sum_{i=1}^{n} w_i g'(\mathbf{v}_i \cdot \mathbf{x}) \mathbf{v}_i + \lambda \nabla\left(\|\mathbf{x}^k\|_2^2\right)].$$

where $\eta > 0$ is the learning rate.

For convenience, we introduce the following notations:

$$\Delta\mathbf{x}^k = \mathbf{x}^{k+1} - \mathbf{x}^k = -\eta E_{\mathbf{x}}(\mathbf{x}^k) \tag{7}$$

$$G^k = G(\mathrm{vx}^k) \tag{8}$$

$$\psi^k = G^{k+1} - G^k \tag{9}$$

3 Main results and proofs

For any $\mathbf{x} \in \mathbf{R}^p$, we write $\| x \| = \sqrt{\sum_{j=1}^{p} (x_j)^2}$, where $\|\cdot\|$ stands for the Euclidean norm in $\mathbf{R}^p$. Let $\Omega_0 = \{\mathbf{x} \in \Omega : E_{\mathbf{x}}(\mathbf{x}) = 0\}$ be the stationary point set of the error function $E(\mathbf{x})$, where $\Omega \subset \mathbf{R}^p$ is a bounded open set. Let $\Omega_{0,s} \subset \mathbf{R}$ be the projection of Ω_0 onto the sth coordinate axis, that

$$\Omega_{0,s} = \left\{ x_s \in R : \mathbf{x} = (x_1, \cdots, x_s, \cdots, x_p)^T \in \Omega_0 \right\} \tag{10}$$

for $s = 1, 2, \cdots, p$. To analyze the convergence of the algorithm, we need the following assumptions:

($A1$) The activation and function g continuously differentiable, $g'(t)$ is uniformly bounded and *Lipschitz* continuous on $\mathbf{R}$;

($A2$) The weight sequence $\left\{\mathbf{w}^k, \mathbf{V}^k\right\}_{k=0}^{\infty}$ is uniformly bounded;

($A3$) The initial input vector of inverse iterative algorithms with L_2 penalty $\mathbf{x}^0$ is uniformly bounded;

($A4$) $\Omega_{0,s}$ does not contain any interior point for every $s = 1, 2, \cdots, p$.

We first present two useful lemmas for the convergence analysis.

Lemma 1. Let $q(x)$ be a function defined on a bounded closed interval $[a,b]$ such that $q'(x)$ is *Lipschitz* continuous with *Lipschitz* constant $K > 0$. Then, $q'(x)$ is differentiable almost everywhere in $[a,b]$ and

$$|q''(x)| \le K, \quad a.e.[a,b] \tag{11}$$

Moreover, there exists a constant $C > 0$ such that

$$q(x) \le q(x_0) + q'(x_0)(x - x_0) + C(x - x_0)^2, x_0, x \in [a,b]. \tag{12}$$

Proof. Since $q'(x)$ is *Lipschitz* continuous on $[a,b]$, $q'(x)$ is absolutely continuous and the derivative $q''(x)$ exists almost everywhere on $[a,b]$. Hence let x is a derivative point of $q'(x)$ on $[a,b]$,

$$\begin{aligned} |q''(x)| &= \left| \lim_{h \to 0} \frac{q'(x+h) - q'(x)}{h} \right| \\ &= \lim_{h \to 0} \left| \frac{q'(x+h) - q'(x)}{h} \right| \le K \end{aligned} \tag{13}$$

$$|q''(x)| \le K, \quad a.e.[a,b] \tag{14}$$

Using the integral *Taylor* expansion, we deduce that

$$\begin{aligned} q(x) &= q(x_0)+q'(x_0)(x-x_0)+(x-x_0)^2\int_0^1(1-t)q''(x_0+t(x-x_0))dt \\ &\le q(x_0)+q'(x_0)(x-x_0)+(x-x_0)^2\int_0^1 K(1-t)dt \\ &= q(x_0)+q'(x_0)(x-x_0)+C(x-x_0)^2. \\ &\left(C=\frac{K}{2},\ x_0,x\in[a,b]\right) \end{aligned} \tag{15}$$

Lemma 2. Let $\{b_m\}$ be a bounded sequence satisfying $\lim_{m\to\infty}(b_{m+1}-b_m)=0$. Write $\gamma_1=\lim_{n\to\infty}\inf_{m>n} b_m$, $\gamma_2=\lim_{n\to\infty}\sup_{m>n} b_m$: There exists a subsequence $\{b_{i_k}\}$ of $\{b_m\}$ such that $S=\{a\in R: b_{i_k}\to a(k\to\infty)\}$. Then we have

$$S=[\gamma_1,\gamma_2] \tag{16}$$

Proof. It is obvious that $\gamma_1\le\gamma_2$ and $S\subseteq[\gamma_1,\gamma_2]$. If $\gamma_1=\gamma_2$, then $\lim_{m\to\infty} b_m=\gamma_1=\gamma_2$, simply proof $S=[\gamma_1,\gamma_2]$. Let us consider the case $\gamma_1<\gamma_2$ and proceed to prove that $S\supseteq[\gamma_1,\gamma_2]$.

For any $a\in(\gamma_1,\gamma_2)$, there exists $\varepsilon>0$, such that $(a-\varepsilon,a+\varepsilon)\subset(\gamma_1,\gamma_2)$. Noting that $\lim_{m\to\infty}(b_{m+1}-b_m)=0$, we observe that b_m travels to and from between γ_1 and γ_2 with very small pace for all large enough m. Hence, there must be infinite number of points of the sequence $\{b_m\}$ falling into $(a-\varepsilon,\mathrm{a}+\varepsilon)$. This implies $a\in S$ and thus $(\gamma_1,\gamma_2)\subseteq S$ Furthermore $[\gamma_1,\gamma_2]\subseteq S$ immediately leads to $S=[\gamma_1,\gamma_2]$. This completes the proof.

Now, we are ready to prove the monotonicity theorem and convergence theorem.

Theorem 1. (**Monotonicity**) Suppose the conditions $(A1)$, $(A2)$, $A(3)$ are valid, and the learning rate satisfies (29), for any given initial input vector $\mathbf{x}^0\in\square^p$, the error function holds that

$$E\left(\mathbf{x}^{k+1}\right)\le E\left(\mathbf{x}^k\right),\quad k=0,1,2,\cdots. \tag{17}$$

then, there exists $E^*\ge 0$ such that

$$\lim_{k\to\infty} E\left(\mathbf{x}^k\right)=E^*. \tag{18}$$

Proof. By assumption $(A1)$ and **Lemma 1**, let

$$|g(t)|,|g'(t)|,|g''(t)|<\tilde{C}. \tag{19}$$

where $t\in R,\tilde{C}$ is constant. By assumption$(A2)$, let

$$\|\mathbf{w}^k\|\le C_1,\|\mathbf{v}_i{}^k\|\le C_2. \tag{20}$$

$i=1,2,\cdots,n,C_1,C_2$ is constant.
there holds

$$\begin{aligned}\|\psi^k\|&=\|G^{k+1}-G^k\|\\&=\sqrt{\sum_{i=1}^{n}(g(\mathbf{v}_i\cdot\mathbf{x}^{k+1})-g(\mathbf{v}_i\cdot\mathbf{x}^k))^2}\\&=\sqrt{\sum_{i=1}^{n}(g'(t_i))^2\|\mathbf{v}_i\|^2\|\Delta\mathbf{x}^k\|^2}\\&\le\sqrt{n}\max_{1\le i\le n}g'(t_i)\|\mathbf{v}_i\|\ \ \|\Delta\mathbf{x}^k\|\\&\le\sqrt{n}\tilde{C}C_2\|\Delta\mathbf{x}^k\|\end{aligned} \tag{21}$$

where $t_i=\mathbf{v}_i\cdot\Delta\mathbf{x}^{k+1}+\theta_i\mathbf{v}_i\cdot\Delta\mathbf{x}^k,\theta_i\in(0,1).$
Using the integral *Taylor* expansion, we deduce that

$$\begin{aligned}&E\left(\mathbf{x}^{k+1}\right)-E\left(\mathbf{x}^k\right)\\&=\left[\tilde{g}\left(\mathbf{w}\cdot G\left(\mathbf{V}\mathbf{x}^{k+1}\right)\right)-\tilde{g}\left(\mathbf{w}\cdot G\left(\mathbf{V}\mathbf{x}^k\right)\right)\right]\\&\quad+\lambda\left[\left\|\mathbf{x}^{k+1}\right\|_2^2-\left\|\mathbf{x}^k\right\|_2^2\right]\\&=\tilde{g}'\left(\mathbf{w}\cdot G\left(\mathbf{V}\mathbf{x}^k\right)\right)\mathbf{w}\cdot\psi^k\\&\quad+\left(\mathbf{w}\cdot\psi^k\right)^2\int_0^1(1-t)\tilde{g}''\left(\mathbf{w}\cdot G\left(\mathbf{V}\mathbf{x}^k\right)+t\psi^k\right)dt\\&\quad+\lambda\sum_{j=1}^{p}[(x_j^{k+1})^2-(x_j^k)^2]\\&=\delta_1+\delta_2+\delta_3.\end{aligned} \tag{22}$$

where,

$$\begin{aligned}\delta_1 &= \tilde{g}'\left(\mathbf{w}\cdot G\left(\mathbf{V}\mathbf{x}^k\right)\right)\mathbf{w}\cdot\psi^k \\ &= \tilde{g}'\left(\mathbf{w}\cdot G\left(\mathbf{V}\mathbf{x}^k\right)\right)\sum_{i=1}^{n} w_i(\mathrm{g}(\mathbf{v}_i\cdot\mathbf{x}^{k+1})-\mathrm{g}(\mathbf{v}_i\cdot\mathbf{x}^k)) \\ &= \sum_{i=1}^{n}\tilde{g}'\left(\mathbf{w}\cdot G\left(\mathbf{V}\mathbf{x}^k\right)\right)w_i\,\mathrm{g}'(\mathbf{v}_i\cdot\mathbf{x}^k)\mathbf{v}_i\cdot\Delta\mathbf{x}^k \\ &+\sum_{i=1}^{n}\tilde{g}'\left(\mathbf{w}\cdot G\left(\mathbf{V}\mathbf{x}^k\right)\right)w_i\left(\int_0^1(1-t)g''\left(\mathbf{v}_i\cdot\mathbf{x}^k+t\mathbf{v}_i\cdot\Delta\mathbf{x}^k\right)dt\right)\left(\mathbf{v}_i\cdot\Delta\mathbf{x}^k\right)^2\end{aligned} \tag{23}$$

According to (6), (7), we can deduce that

$$\begin{aligned}\delta_1 &= -\frac{1}{\eta}\Delta\mathbf{x}^k\cdot\Delta\mathbf{x}^k-\lambda\nabla\left(\|f(\mathbf{x})\|_2^2\right)\cdot\Delta\mathbf{x}^k \\ &= -\frac{1}{\eta}\left\|\Delta\mathbf{x}^k\right\|^2-2\lambda\sum_{j=1}^{p}x_j^k\Delta\mathrm{x}_j^k\end{aligned} \tag{24}$$

It follows from (19) and (20) that

$$\delta_1 \le -\frac{1}{\eta}\left\|\Delta\mathbf{x}^k\right\|^2-2\lambda\sum_{j=1}^{p}x_j^k\Delta\mathrm{x}_j^k+A_1\|\ \Delta\mathbf{x}^k\|^2 \tag{25}$$

where $A_1=\frac{1}{2}\tilde{C}^2C_1C_2^2$.

$$\begin{aligned}\delta_2 &= \left(\mathbf{w}\cdot\psi^k\right)^2\int_0^1(1-t)\tilde{g}''\left(\mathbf{w}\cdot G\left(\mathbf{V}\mathbf{x}^k\right)+t\psi^k\right)dt \\ &\le \|\ \mathbf{w}\|^2\|\ \psi^k\|^2A_2' \\ &\le A_2''\|\ \psi^k\|^2\end{aligned} \tag{26}$$

where $A_2'=\frac{1}{2}\tilde{C},A_2''=A_2'C_1^2$.

Employing (21), we deduce that

$$\delta_2 \le A_2\|\ \Delta\mathbf{x}^k\|^2 \tag{27}$$

where $A_2=n\tilde{C}^2C_2^2A_2''$.

Indeed

$$\begin{aligned}\delta_3 &= \lambda\sum_{j=1}^{p}\left(\left(x_j^{k+1}\right)^2 - \left(x_j^k\right)^2\right)\\ &= 2\lambda\sum_{j=1}^{p} x_j^k \Delta x_j^k + 2\lambda\sum_{j=1}^{p}(\Delta x_j^k)^2\\ &= 2\lambda\sum_{j=1}^{p} x_j^k \Delta x_j^k + 2\lambda\left\|\Delta \mathrm{x}^k\right\|^2\end{aligned} \tag{28}$$

Let

$$\eta < \frac{1}{A_1 + A_2 + 2\lambda} \tag{29}$$

we have that,

$$\begin{aligned}&E\left(\mathbf{x}^{k+1}\right) - E\left(\mathbf{x}^k\right) = \delta_1 + \delta_2 + \delta_3\\ &\leq -\frac{1}{\eta}\left\|\Delta\mathbf{x}^k\right\|^2 - 2\lambda\sum_{j=1}^{p} x_j^k \Delta x_j^k + A_1\left\|\Delta\mathbf{x}^k\right\|^2 +\\ &A_2\left\|\Delta\mathbf{x}^k\right\|^2 + 2\lambda\sum_{j=1}^{p} x_j^k \Delta x_j^k + 2\lambda\left\|\Delta\mathbf{x}^k\right\|^2\\ &= (-\frac{1}{\eta} + A_1 + A_2 + 2\lambda)\left\|\Delta\mathbf{x}^k\right\|^2 < 0.\end{aligned} \tag{30}$$

The proof of the monotonicity theorem is completed.

Theorem 2. (**Boundedness**) The iterative sequence of input vectors $\left\{X^k\right\}_{k=0}^{\infty}$ of inverse iterative algorithms with L_2 penalty is uniformly bounded.

Proof. By assumption $(A3)$, let

$$\left\|\mathbf{x}^0\right\| \leq M_0 \tag{31}$$

According to (6); (7), we can deduce that

$$\mathbf{x}^1 = \mathbf{x}^0 - \eta E_{\mathbf{x}^0}\left(\mathbf{x}^0\right) \tag{32}$$

$$\mathbf{x}^1 = \mathbf{x}^0 - \eta[\tilde{g}'\left(\mathbf{w}\cdot G\left(\mathbf{V}\mathbf{x}^0\right)\right)\sum_{i=1}^{n} w_i g'\left(\mathbf{v}_i \cdot \mathbf{x}^0\right)\mathbf{v}_i + \lambda\nabla\left(\left\|\mathbf{x}\right\|_2^2\right) \tag{33}$$

By (29), let $\eta < \dfrac{1}{A_1 + A_2 + 2\lambda} < \dfrac{1}{2\lambda}$, then we have

$$\left\|\mathbf{x}^1\right\| \le \left\|\mathbf{x}^0\right\| + \frac{1}{2\lambda}[\mathrm{C}_4 + 2\lambda M_0] \equiv M_1 \tag{34}$$

where $C_4 = \sup \tilde{g}'\left(\mathbf{w}\cdot G\left(\mathbf{V}\mathbf{x}^0\right)\right) \sup\limits_{t\in R} g'(t) \sup\left\|\mathrm{w}^0\right\| \sup\left\|\mathrm{v}_i^0\right\|$.

In the same way, we can deduce that

$$\left\|\mathbf{x}^2\right\| \le M_1 + (\mathrm{C}_4 + 2\lambda M_1) \equiv M_2 \tag{35}$$

Repeating this procedure, There is constant $M_j (3 \le j \le k)$, such that

$$\left\|\mathbf{x}^j\right\| \le M_j \tag{36}$$

Let. $M = \max\left\{M_0, M_1, \cdots, M_K\right\}$, then $\left\|\mathbf{x}^j\right\| \le M$.

Hence,

$$\left\|\mathbf{x}^k\right\| \le M, k = 0, 1, \cdots \tag{37}$$

The iterative sequence of input vectors $E(\mathbf{x}^0)$ is uniformly bounded.

Theorem 3. (**Weak convergence**) Assume that conditions $(A1)$, $(A2)$, $(A3)$ are valid, the error function defined by (3), and the learning rate satisfies (30), for any arbitrary initial value $\mathbf{x}^0 \in □^{p}, \mathbf{x}^k$ defined by (6), then

$$\lim_{k\to\infty} \left\|E_{\mathrm{x}}\left(\mathbf{x}^k\right)\right\| = 0 \tag{38}$$

Proof. By the results of **Theorem 2**. Let $\alpha = \dfrac{1}{\eta} - A_4 - \lambda A_3$,

$$\begin{aligned} E\left(\mathbf{x}^{k+1}\right) &\le E\left(\mathbf{x}^k\right) - \alpha\left\|\Delta\mathbf{x}^k\right\|^2 \\ &\le E\left(\mathbf{x}^{k-1}\right) - \alpha\left(\left\|\Delta\mathbf{x}^k\right\|^2 + \left\|\Delta\mathbf{x}^{k-1}\right\|^2\right) \\ &\le \cdots \le E\left(\mathbf{x}^0\right) - \alpha\sum_{l=0}^{k}\left\|\Delta\mathbf{x}^k\right\|^2. \end{aligned} \tag{39}$$

For $E(\mathbf{x}^k)\ge 0, k\in\square$, where $K\in\square^+$, then

$$\alpha\sum_{k=0}^{K}\left\|\Delta\mathbf{x}^k\right\|^2\le E\left(\mathbf{x}^0\right). \tag{40}$$

Let $K\to\infty$, we can deduce that

$$\sum_{k=0}^{\infty}\left\|\Delta\mathbf{x}^k\right\|^2\le\frac{1}{\alpha}E\left(\mathbf{x}^0\right)<\infty \tag{41}$$

This immediately gives

$$\lim_{k\to\infty}\|\ \Delta\mathbf{x}^k\|=0. \tag{42}$$

Combining (6) and (8) results in:

$$\lim_{k\to\infty}\left\|E_{\mathbf{x}}\left(\mathbf{x}^k\right)\right\|=0. \tag{43}$$

This completes the proof of the weak convergence.

Theorem 4. (**Strong convergence**) If assumptions $(A1)-(A4)$ are valid, there holds the strong convergence: There exists $\mathbf{x}^*\in\Omega_0$ such that

$$\lim_{k\to\infty}\mathbf{x}^k=\mathbf{x}^*. \tag{44}$$

Proof. According to $(A3)$, the sequence $\left\{\mathbf{x}^k\right\}\left(k\in\square\right)$has a subsequence that is convergent to, say, $\mathbf{x}^*$. Then $\mathbf{x}^{k_i}\to\mathbf{x}^*(k_i\to\infty)$; k_i is a subsequence of k. It follows from (43) and the continuity of $E_{\mathbf{x}}\left(\mathbf{x}\right)$ that

$$\left\|E_{\mathbf{x}}\left(\mathbf{x}^*\right)\right\|=\lim_{i\to\infty}\left\|E_{\mathbf{x}}\left(\mathbf{x}^{k_i}\right)\right\|=\lim_{m\to\infty}\left\|E_{\mathbf{x}}\left(\mathbf{x}^k\right)\right\|=0. \tag{45}$$

This implies that $\mathbf{x}^*$ is a stationary point of $E\left(\mathbf{x}\right)$. Hence, $\left\{\mathbf{x}^k\right\}$ has at least one accumulation point and every accumulation point must be a stationary point of $E\left(\mathbf{x}\right)$.

Next, by reduction to absurdity, we prove that $\left\{\mathbf{x}^k\right\}$ has precisely one accumulation point. Let us assume to the contrary that $\left\{\mathbf{x}^k\right\}$ has at least two accumulation points, without loss of generality, we assume that the first components of x andex do not equal to each other, that is, $\overline{x}_1\ne\tilde{x}_1$. For

$\forall\lambda \in (0,1)$, let $x_1' = \lambda\bar{x}_1 + (1-\lambda)\tilde{x}_1$. By **Lemma 2**, there exists a subsequence $\left\{x_1^{k_{i_1}}\right\} \subset \{x_1^k\}$ such that $x_1^{k_{i_1}} \to x_1', (i_1 \to \infty)$, here fki1g is a subsequence of fkg. Due to the boundedness of $\left\{\mathbf{x}^{k_{i_1}}\right\}$, there is a convergent subsequence $\left\{k_{i_2}\right\} \subset \left\{k_{i_1}\right\}$, we define $\mathbf{x}^{k_{i_2}} \to \mathbf{x}_2'(i_2 \to \infty)$. Repeating this procedure, we end up with decreasing subsequences $x_t^{k_{i_t}} \to x_t', (i_t \to \infty), t = 1, 2, \cdots, p$. Write $x' = \{x_1', x_2', \cdots, x_p'\}$. Then, we see that $x'$is an accumulation point of $\{\mathbf{x}_k\}$for any $\lambda \in (0,1)$. But this means that 0 has interior points, which contradicts with the assumption $\left(A4\right)$. Thus, $\mathbf{x}^*$must be a unique accumulation point of $\mathbf{x}^k$. This completes the proof of the strong convergence.

4 Conclusions

An inverse iterative algorithm for neural networks with L_2 penalty has been proposed in this paper. The main contributions of this paper are focused on the theoretical analyses. The monotonicity of the error function and boundedness of inputs have been proved under mild conditions. The gradient sequence of the error function with respect to the inputs tends to zero as the iterations go to infinity, this results in the weak convergence. The strong convergence (the input sequence approaches a fixed point) is then obtained by an additional assumption.

Acknowledgments

The authors wish to thank the anonymous reviewers for careful error proofing of the manuscript and many insightful comments and suggestions which greatly improved this work. This project was supported in part by the National Natural Science Foundation of China (No. 61305075), the China Postdoctoral Science Foundation (No. 2012M520624), Natural Science Foundation of Shandong Province (No. ZR2013FQ004), the Specialized Research Fund for the Doctoral Program of Higher Education of China (No. 20130133120014) and the Fundamental Research Funds for the Central Universities (No. 15CX05053A, 15CX08011A).

References

1. J. M. Zurada, *Introduction to artificial neural systems: West Publishing Company,* St. Paul, 1992.
2. S. S. Haykin, *Neural networks and learning machines: Pearson Education*, Upper Saddle River, 2009.
3. P.Werbos, *Beyond regression: New tools for prediction and analysis in the behavioral sciences,* 1974.
4. D. E. Rumelhart, G. E. Hinton, and R. J. Williams, *Learning representations by backpropagating errors,* Cognitive modeling, 1988.
5. G. E. Hinton, *Connectionist learning procedures,* Artificial intelligence, vol. 40, no. 1, pp. 185-234, 1989.
6. R. Reed, *Pruning algorithms-a survey,* Neural Networks, IEEE Transactions on, vol. 4, no. 5, pp. 740-747, 1993.
7. M. Ishikawa, *Structural learning with forgetting,* Neural Networks, vol. 9, no. 3, pp. 509-521, 1996.
8. R. Setiono, *A penalty-function approach for pruning feedforward neural networks,* Neural computation,vol. 9, no. 1, pp. 185-204, 1997.
9. H. M. Shao,W.Wei., and L.-j. Liu, “Convergence of Online Gradient Method with Penalty for BP Neural Networks,” Communications in Mathematical Research vol. 26, no. 1, pp. 67-75, 2010.
10. J.Wang, J. Yang, andW.Wu, *Convergence of cyclic and almost-cyclic learning with momentum for feedforward neural networks,* Neural Networks, IEEE Transactions on, vol. 22, no. 8, pp. 1297-1306, 2011.
11. G. Uhlmann, *Inside out: inverse problems and applications*: Cambridge University Press, 2003.
12. M. Zamparo, S. Stramaglia, J. Banavar, and A. Maritan, *Inverse problem for multivariate time series using dynamical latent variables,* Physica A: Statistical Mechanics and its Applications, vol. 391, no. 11, pp. 3159-3169, 2012.
13. J. Kindermann, and A. Linden, *Inversion of neural networks by gradient descent,* Parallel computing,vol. 14, no. 3, pp. 277-286, 1990.
14. A. Fanni, and A. Montisci, *A neural inverse problem approach for optimal design,* Magnetics, IEEE Transactions on, vol. 39, no. 3, pp. 1305-1308, 2003.
15. Y. Hayakawa, and K. Nakajima, *Design of the inverse function delayed neural network for solving combinatorial optimization problems,* Neural Networks, IEEE Transactions on, vol. 21, no. 2, pp. 224-237, 2010.
16. D. Cherubini, A. Fanni, A. Montisci, and P. Testoni, *Inversion of MLP neural networks for direct solution of inverse problems,* Magnetics, IEEE Transactions on, vol. 41, no. 5, pp. 1784-1787, 2005.
17. E. W. Saad, and D. C. Wunsch II, *Neural network explanation using inversion,* Neural Networks,vol. 20, no. 1, pp. 78-93, 2007.

18. R. W. Duren, R. J. Marks, P. D. Reynolds, and M. L. Trumbo, *Real-time neural network inversion on the SRC-6e reconfigurable computer,* Neural Networks, IEEE Transactions on, vol. 18, no. 3, pp. 889-901, 2007.
19. S.-q. Meng, *Convergence of an inverse iteration algorithm for neural networks,* Dalian University of Technology, 2007.
20. Z.-b. Xu, H. Zhang, Y. Wang, X.-y. Chang, and Y. Liang, *L 1/2 regularization,* Science China-Information Sciences, vol. 53, no. 6, pp. 1159-1169, 2010.
21. W.Wu, Q. Fan, J. M. Zurada, J.Wang, D. Yang, and Y. Liu, *Batch gradient method with smoothing L1/2 regularization for training of feedforward neural networks,* Neural Networks, vol. 50, pp. 72-78, 2014.