3

Discrete-Time Adaptive Neural Backstepping

This chapter deals with adaptive tracking for a class of MIMO discrete-time nonlinear systems in presence of bounded disturbances. In this chapter, a high order neural network structure is used to approximate a control law designed by the backstepping technique, applied to a block strict feedback form (BSFF). It also presents the respective stability analysis, on the basis of the Lyapunov approach, for the whole scheme including the extended Kalman filter (EKF)-based NN learning algorithm. Applicability of this scheme is illustrated via simulation for a discrete-time nonlinear model of an electric induction motor.

In recent adaptive and robust control literature, numerous approaches have been proposed for the design of nonlinear control systems. Among these, adaptive backstepping constitutes a major design methodology [6, 9]. The idea behind backstepping design is that some appropriate functions of state variables are selected recursively as virtual control inputs for lower dimension subsystems of the overall system [12]. Each backstepping stage results in a new virtual control designs from the preceding design stages. When the procedure ends, a feedback design for the true control input results, which achieves the original design objective. The backstepping technique provides a systematic framework for the design of tracking and regulation strategies, suitable for a large class of state feedback linearizable nonlinear systems [1, 9–11].

3.1 Neural Backstepping Controller Design

The model of many practical nonlinear systems can be expressed in (or transformed into) a special state-space form named block strict feedback form (BSFF) [9] as follows:

$$\begin{aligned} x^i(k+1) &= f^i(\overline{x}^i(k)) + g^i(\overline{x}^i(k))x^{i+1}(k) + d^i(k), \qquad i = 1, 2, \ldots, r-1, \\ x^r(k+1) &= f^r(x(k)) + g^r(x(k))\,u(k) + d^r(k), \\ y(k) &= x^1(k), \end{aligned} \tag{3.1}$$

E.N. Sanchez et al.: *Discrete-Time High Order Neural Control – Trained with Kalman Filtering*, Studies in Computational Intelligence (SCI) **112**, 11–28 (2008)
www.springerlink.com

where $x(k) = \left[x^{1\top}(k), \ldots, x^{r\top}(k)\right]^{\top}$ are the state variables, and $\overline{x}^i(k) = \left[x^{1\top}, x^{2\top}, \ldots, x^{i\top}\right]^{\top}$, $x^i \in \Re^{n_i}$, $r \geq 2$, r is the number of blocks, $u(k) \in \Re^m$ is the system input, $y(k) \in \Re^m$ is the system output; for simplicity of notation through the remaining of this chapter $d^i(k) = d^i(x(k), k) \in \Re^{n_i}$ is the bounded unknown disturbance vector, then there exists a constant $\overline{d}_i$ such that $\|d_i(k)\| \leq \overline{d}_i$, for $0 < k < \infty$, $f^i(\bullet)$ and $g^i(\bullet)$ are unknown smooth nonlinear functions.

If we consider the original system (3.1) as a one-step ahead predictor, then we can transform it into an equivalent maximum r-step ahead one, which can predict the future states $x^1(k+r)$, $x^2(k+r-1), \ldots, x^r(k+1)$; the causality contradiction is avoided when the controller is constructed based on the maximum r-step ahead prediction by backstepping [3, 4]:

$$\begin{aligned}
x^1(k+r) &= \overline{f}^1\left(\bar{x}^1(k)\right) + \overline{g}^1\left(\overline{x}^1(k)\right) x^2(k+r-1) + d^1(k+r),\\
&\vdots\\
x^{r-1}(k+2) &= \overline{f}^{r-1}\left(\overline{x}^{r-1}(k)\right) + \overline{g}^{r-1}\left(\overline{x}^{r-1}(k)\right) x^r(k+1) + d^{r-1}(k+2),\\
x^r(k+1) &= \overline{f}^r(x(k)) + \overline{g}^r(x(k))\, u(k) + d^r(k),\\
y(k) &= x^1(k),
\end{aligned} \tag{3.2}$$

where $\overline{f}^i(\bullet)$ and $\overline{g}^i(\bullet)$ are unknown functions of $f^i\left(\overline{x}^i(k)\right)$ and $g^i(\overline{x}^i(k))$, respectively. For convenience of analysis, let us define $(i = 1, \ldots, r-1)$

$$\begin{aligned}
\overline{f}^i(k) &\triangleq \overline{f}^i\left(\overline{x}_i(k)\right),\\
\overline{g}^i(k) &\triangleq \overline{g}^i\left(\overline{x}_i(k)\right),\\
\overline{f}^r(k) &\triangleq \overline{f}^r\left(X(k)\right),\\
\overline{g}^r(k) &\triangleq \overline{g}^r\left(X(k)\right).
\end{aligned}$$

Then, system (3.2) can be written as (for details please see *Appendix A*)

$$\begin{aligned}
x^1(k+r) &= \overline{f}^1(k) + \overline{g}^1(k) x^2(k+r-1) + d^1(k+r),\\
&\vdots\\
x^{r-1}(k+2) &= \overline{f}^{r-1}(k) + \overline{g}^{r-1}(k) x^r(k+1) + d^{r-1}(k+2),\\
x^r(k+1) &= \overline{f}^r(k) + \overline{g}^r(k) u(k) + d^r(k),\\
y(k) &= x_1(k).
\end{aligned} \tag{3.3}$$

The objective is to design a control $u(k)$ to force the system output $y(k)$ to track a desired trajectory $y_{\mathrm{d}}(k)$. Once (3.3) is defined, we apply the well known backstepping technique [9]. For system (3.2), we can define the desired virtual controls $(\alpha^{j*}(k),\ j = 1, \ldots, r-1)$ and the ideal practical control $(u^*(k))$ as follows:

$$\begin{aligned}
\alpha^{1*}(k) &\triangleq x^2(k) = \varphi^1\big(\overline{x}^1(k), y_{\mathrm{d}}\,(k+r)\big),\\
\alpha^{2*}(k) &\triangleq x^3(k) = \varphi^2\big(\overline{x}^2(k), \alpha^{1*}(k)\big),\\
&\vdots\\
\alpha^{r-1*}(k) &\triangleq x^r(k) = \varphi^{r-1}\left(\overline{x}^{r-1}(k), \alpha^{r-2*}(k)\right),\\
u^*(k) &= \varphi^r\big(x(k), \alpha^{r-1*}(k)\big),\\
y(k) &= x^1(k),
\end{aligned} \tag{3.4}$$

where $\varphi^j(j = 1, \ldots, r)$ are nonlinear smooth functions. It is obvious that the desired virtual controls $\alpha^{i*}(k)$ and the ideal control $u^*(k)$ will drive the output $y(k)$ to track the desired signal $y_{\mathrm{d}}(k)$ only if the exact system model is known and there are no unknown disturbances. However, in practical applications, these two conditions cannot be satisfied. In the following, neural networks will be used to approximate the desired virtual controls, as well as the desired practical controls, when the conditions established above are not satisfied. As in [4], we construct the virtual and practical controls via embedded backstepping without the causality contradiction [3]. Let us approximate the virtual controls and practical control by the following HONN ($i = 1, \ldots, r-1$):

$$\begin{aligned}
\alpha^i(k) &= {w^i}^\top z^i(\varrho^i(k)), \qquad i = 1, \ldots, r-1,\\
u(k) &= {w^r}^\top z^r(\varrho^r(k)),
\end{aligned} \tag{3.5}$$

with

$$\begin{aligned}
\varrho^1(k) &= \left[x^1(k), y_{\mathrm{d}}(k+r)\right]^\top,\\
\varrho^i(k) &= \left[\overline{x}^i(k), \alpha^{i-1}(k)\right]^\top, \qquad i = 2, \ldots, r-1,\\
\varrho^r(k) &= \left[x(k), \alpha^{r-1}(k)\right]^\top,
\end{aligned}$$

where $w^j \in \Re^{L_j}$ are the estimates of ideal constant weights $w^{j*}(j = 1, \ldots, r)$ and $z^j \in \Re^{L_j \times n_j}$. Define the estimation error as

$$\widetilde{w}^j(k) = w^{j*} - w^j(k). \tag{3.6}$$

Using the ideal constant weights and from (2.11) it follows that there exists a HONN, which approximate the virtual controls and practical control with a minimal error, defined as

$$\begin{aligned}
\alpha^i(k) &= {w^{i*}}^\top z^i(\varrho^i(k)),\\
u(k) &= {w^{r*}}^\top z^r(\varrho^r(k)) + \epsilon_{z^i}, \qquad i = 1, \ldots, r-1.
\end{aligned} \tag{3.7}$$

Then the corresponding weights updating laws are defined by

$$w^j(k+1) = w^j(k) + \eta^j K^j(k) e^j(k), \tag{3.8}$$

with

$$\begin{aligned} K^j(k) &= P^j(k) H^j(k) M^{j^{-1}}(k), \\ M^j(k) &= R^j(k) + H^{j^\top}(k) P^j(k) H^j(k), \\ P^j(k+1) &= P^j(k) - K^j(k) H^{j^\top}(k) P^j(k) + Q^j(k), \\ H^j(k) &= \left[\frac{\partial \widehat{v}^j(k)}{\partial w^j(k)} \right], \end{aligned} \tag{3.9}$$

and

$$e^j(k) = v^j(k) - \widehat{v}^j(k), \tag{3.10}$$

where $v^i(k) \in \Re^{n_i}$ is the desired signal and $\widehat{v}^i(k) \in \Re^{n_i}$ is the HONN function approximation defined, respectively, as follows

$$\begin{aligned} v^1(k) &= y_{\mathrm{d}}(k), \\ v^2(k) &= x^2(k), \\ &\vdots \\ v^r(k) &= x^r(k) \end{aligned} \tag{3.11}$$

and

$$\begin{aligned} \widehat{v}^1(k) &= y(k), \\ \widehat{v}^2(k) &= \alpha^1(k), \\ &\vdots \\ \widehat{v}^r(k) &= \alpha^{r-1}(k), \end{aligned} \tag{3.12}$$

$e^j(k)$ denotes the error at each step, as

$$\begin{aligned} e^1(k) &= y_{\mathrm{d}}(k) - y(k), \\ e^2(k) &= x^2(k) - \alpha^1(k), \\ &\vdots \\ e^r(k) &= x^r(k) - \alpha^{r-1}(k). \end{aligned} \tag{3.13}$$

The proposed control scheme is shown in Fig. 3.1. Besides, it is worth to include the following comments:

Comment 3.1. The NN approximation error vector ϵ_z is bounded. This is a well known neural network property [2].

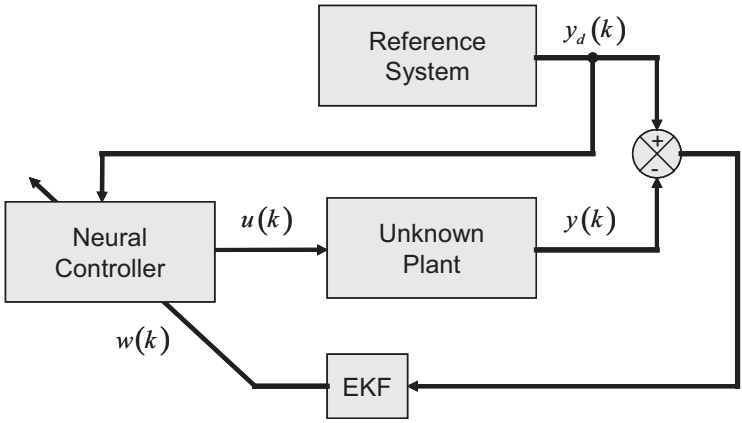

Fig. 3.1. Neural backstepping control scheme

Comment 3.2. The gain matrix of the EKF ($K(k)$) is bounded by a constant $\overline{K} > 0$, that is, $\|K(k)\| \leq \overline{K}$.

Before proceeding to demonstrate the main result of this chapter, we need to establish the following two lemmas.

Lemma 3.1. *The dynamics of the tracking error* (3.10) *can be formulated as*

$$e^j(k+1) = e^j(k) + \Delta e^j(k), \qquad (1 \leq j \leq r)\,, \tag{3.14}$$

with $\Delta e^j(k) \leq -\gamma^j e^j(k)$ *and* $\gamma^j = \max \left\| H^{j^\top}(k)\eta^j K^j(k) \right\|$.

Proof. Using (3.10) and considering that $\upsilon(k)$ do not depend on the HONN parameters, we obtain

$$\frac{\partial e^i(k)}{\partial w^i(k)} = -\frac{\partial \widehat{\upsilon}(k)}{\partial w^i(k)}. \tag{3.15}$$

Let us approximate (3.15) by

$$\Delta e^i(k) = \left[\frac{\partial e^i(k)}{\partial w^i(k)}\right]^\top \Delta w^i(k). \tag{3.16}$$

Substituting (3.9) and (3.15) in (3.16) yields

$$\Delta e^i(k) = -H^{i^\top}(k)\eta^i K^i(k) e^i(k). \tag{3.17}$$

Define

$$\gamma^i = \max \left\| H^{i^\top}(k)\eta^i K^i(k) \right\|$$

then we have

$$\Delta e^i(k) \leq -\gamma^i e^i(k). \tag{3.18}$$

□

Considering (3.1)–(3.13), we establish the main result of this chapter in the following theorem.

Theorem 3.1. *For the system* (3.1)*, the HONN* (3.5) *trained with the EKF-based algorithm* (3.9) *to approximate the control law* (3.4) *ensures that the tracking error* (3.13) *is semiglobally uniformly ultimately bounded (SGUUB); moreover, the HONN weights remain bounded.*

Proof. For the first block of system (3.1), with the virtual control $\alpha^{1*}(k)$ approximated by the HONN $\left(\alpha^1(k) = w^{1^\top} z^1\left(\varrho^1(k)\right)\right)$ and $e^1(k)$ defined as in (3.13), consider the Lyapunov function candidate

$$V^1(k) = e^{1^\top}(k)e^1(k) + \widetilde{w}^{1^\top}(k)\widetilde{w}^1(k), \tag{3.19}$$

whose first difference is

$$\begin{aligned}\Delta V^1(k) &= V^1(k+1) - V^1(k),\\ &= e^{1^\top}(k+1)e^1(k+1) + \widetilde{w}^{1^\top}(k+1)\widetilde{w}^1(k+1)\\ &\quad - e^{1^\top}(k)e^1(k) - \widetilde{w}^{1^\top}(k)\widetilde{w}^1(k).\end{aligned} \tag{3.20}$$

From (3.6) and (3.8), then

$$\widetilde{w}^1(k+1) = \widetilde{w}^1(k) - \eta^1 K^1(k)e^1(k). \tag{3.21}$$

Let us define

$$\begin{aligned}&\left[\widetilde{w}^1(k) - \eta^1 K^1(k)e^1(k)\right]^\top \left[\widetilde{w}^1(k) - \eta^1 K^1(k)e^1(k)\right]\\ &= \widetilde{w}^{1^\top}(k)\widetilde{w}^1(k) - 2\widetilde{w}^{1^\top}(k)\eta^1 K^1(k)e^1(k)\\ &\quad + \left(\eta^1 K^1(k)e^1\right)^\top \eta^1 K^1(k)e^1(k).\end{aligned} \tag{3.22}$$

From (3.13), then

$$\begin{aligned}e^1(k+1) &= e^1(k) + \Delta e^1(k),\\ e^{1^\top}(k+1)e^1(k+1) &= e^{1^\top}(k)e^1(k) + e^{1^\top}(k)\Delta e^1(k)\\ &\quad + \Delta e^{1^\top}(k)e^1(k) + \Delta e^{1^\top}(k)\Delta e^1(k),\\ e^{1^\top}(k+1)e^1(k+1) - e^{1^\top}(k)e^1(k) &= e^{1^\top}(k)\Delta e^1(k) + \Delta e^{1^\top}(k)e^1(k)\\ &\quad + \Delta e^{1^\top}(k)\Delta e^1(k),\end{aligned}$$

where $\Delta e^1(k)$ is the error difference. Substituting (3.21) and (3.22) in (3.20) results in

$$\begin{aligned}\Delta V^1(k) &= e^{1^\top}(k)\Delta e^1(k) + \Delta e^{1^\top}(k)e^1(k) + \Delta e^{1^\top}(k)\Delta e^1(k)\\ &\quad - 2\widetilde{w}^{1^\top}(k)\eta^1 K^1(k)e^1(k)\\ &\quad + (\eta^1 K^1(k)e^1(k))^\top \eta^1 K^1(k)e^1(k).\end{aligned} \tag{3.23}$$

From Lemma 3.1, substituting (3.18), we obtain

$$\begin{aligned}\Delta V^1(k) \leq & -2\gamma^1 e^{1^\top}(k)e^1(k) + \gamma^{1^2} e^{1^\top}(k)e^1(k) - 2\widetilde{w}^{1^\top}(k)\eta^1 K^1(k)e^1(k) \\ & + (\eta^1 K^1(k)e^1(k))^\top \eta^1 K^1(k)e^1(k), \\ \leq & -2\gamma^1 \left\|e^1(k)\right\|^2 + \gamma^{1^2}\left\|e^1(k)\right\|^2 - 2\|\eta^1 K^1(k)\|\left\|\widetilde{w}^1(k)\right\|\left\|e^1(k)\right\| \\ & + \|\eta^1 K^1(k)\|^2\left\|e^1(k)\right\|^2, \\ \leq & -2\gamma^1 \left\|e^1(k)\right\|^2 + \gamma^{1^2}\left\|e^1(k)\right\|^2 \\ & - 2\|\eta^1 K^1(k)\|\left\|w^* - w^1_{\max}\right\|\left\|e^1(k)\right\| \\ & + \|\eta^1 K^1(k)\|^2\left\|e^1(k)\right\|^2, \end{aligned} \tag{3.24}$$

with $\gamma^1 = \max\left\|H^{1^\top}(k)\eta^1 K^1(k)\right\| > 1$. There is $\eta^1 > 0$ such that $\vartheta^1 > 0$ with

$$\vartheta^1 = 2\gamma^1 - \gamma^{1^2} - \|\eta^1 K^1(k)\|^2,$$

then

$$\Delta V^1(k) \leq 0, \quad \text{once} \quad \left\|e^1(k)\right\| > \kappa^1, \tag{3.25}$$

with κ^1 defined as

$$\kappa^1 > \frac{\left(\gamma^{1^2} + \|\eta^1 K^1(k)\|^2\right)\left\|e^1_{\max}\right\|}{2\|\eta^1 K^1(k)\|}.$$

Therefore, the solution of (3.14) and (3.21) is stable, which leads to the SGUUB of $e^1(k)$ and $\widetilde{w}^1(k)$.

For the following ith ($i = 2, \ldots, r-1$) equation of the system (3.1), with the virtual control $\alpha^{i*}(k)$ approximated by the HONN $\alpha^i(k) = w^{i^\top} z^i(\varrho^i(k))$ and $e^i(k)$ defined in (3.13), consider the Lyapunov function candidate

$$V^i(k) = e^{i^\top}(k)e^i(k) + \widetilde{w}^{i^\top}(k)\widetilde{w}^i(k), \tag{3.26}$$

whose first difference is

$$\begin{aligned}\Delta V^i(k) &= V^i(k+1) - V^i(k), \\ &= e^{i^\top}(k+1)e^i(k+1) + \widetilde{w}^{i^\top}(k+1)\widetilde{w}^i(k+1) \\ &\quad - e^{i^\top}(k)e^i(k) - \widetilde{w}^{i^\top}(k)\widetilde{w}^i(k).\end{aligned} \tag{3.27}$$

From (3.6) and (3.8), then

$$\widetilde{w}^i(k+1) = \widetilde{w}^i(k) - \eta^i K^i(k)e^i(k). \tag{3.28}$$

Let us define

$$\begin{aligned}
&\left[\widetilde{w}^i(k) - \eta^i K^i(k) e^i(k)\right]^\top \left[\widetilde{w}^i(k) - \eta^i K^i(k) e^i(k)\right] \\
&= \widetilde{w}^{i^\top}(k)\widetilde{w}^i(k) - 2\widetilde{w}^{i^\top}(k)\eta^i K^i(k) e^i(k) \\
&\quad + \left(\eta^i K^i(k) e^i(k)\right)^\top \eta^i K^i(k) e^i(k).
\end{aligned} \tag{3.29}$$

From (3.13), then

$$\begin{aligned}
e^i(k+1) &= e^i(k) + \Delta e^i(k), \\
e^{i^\top}(k+1)e^i(k+1) &= e^{i^\top}(k)e^i(k) + e^{i^\top}(k)\Delta e^i(k) \\
&\quad + \Delta e^{i^\top}(k)e^i(k) \\
&\quad + \Delta e^{i^\top}(k)\Delta e^i(k), \\
e^{i^\top}(k+1)e^i(k+1) - e^{i^\top}(k)e^i(k) &= e^{i^\top}(k)\Delta e^i(k) + \Delta e^{i^\top}(k)e^i(k) \\
&\quad + \Delta e^{i^\top}(k)\Delta e^i(k),
\end{aligned}$$

where $\Delta e^i(k)$ is the error difference. Substituting (3.28) and (3.29) in (3.27) results in

$$\begin{aligned}
\Delta V^i(k) = {} & e^{i^\top}(k)\Delta e^i(k) + \Delta e^{i^\top}(k)e^i(k) + \Delta e^{i^\top}(k)\Delta e^i(k) \\
& - 2\widetilde{w}^{i^\top}(k)\eta^i K^i(k) e^i(k) \\
& + \left(\eta^i K^i(k) e^i(k)\right)^\top \eta^i K^i(k) e^i(k).
\end{aligned} \tag{3.30}$$

From Lemma 3.1, substituting (3.18), we obtain

$$\begin{aligned}
\Delta V^i(k) &\le -2\gamma^i e^{i^\top}(k)e^i(k) + \gamma^{i^2} e^{i^\top}(k)e^i(k) - 2\widetilde{w}^{i^\top}(k)\eta^i K^i(k) e^i(k) \\
&\quad + \left(\eta^i K^i(k) e^i(k)\right)^\top \eta^i K^i(k) e^i(k), \\
&\le -2\gamma^i \left\|e^i(k)\right\|^2 + \gamma^{i^2}\left\|e^i(k)\right\|^2 - 2\left\|\eta^i K^i(k)\right\| \left\|\widetilde{w}^i(k)\right\| \left\|e^i(k)\right\| \\
&\quad + \left\|\eta^i K^i(k)\right\|^2 \left\|e^i(k)\right\|^2, \\
&\le -2\gamma^i \left\|e^i(k)\right\|^2 + \gamma^{i^2}\left\|e^i(k)\right\|^2 \\
&\quad - 2\left\|\eta^i K^i(k)\right\| \left\|w^i_{\max} - w^*\right\| \left\|e^i(k)\right\| \\
&\quad + \left\|\eta^i K^i(k)\right\|^2 \left\|e^i(k)\right\|^2.
\end{aligned} \tag{3.31}$$

with $\gamma^i = \max\left\|H^{i^\top}(k)\eta^i K^i(k)\right\|$. There is $\eta^i > 0$ such that $\vartheta^i > 0$ with

$$\vartheta^i = 2\gamma^i - \gamma^{i^2} - \left\|\eta^i K^i(k)\right\|^2,$$

then

$$\Delta V^i(k) \le 0, \quad \text{once} \quad \left\|e^i(k)\right\| > \kappa^i, \tag{3.32}$$

with κ^i defined as

$$\kappa^i > \frac{\left(\gamma^{i^2} + \left\|\eta^i K^i(k)\right\|^2\right)\left\|e^i_{\max}\right\|}{2\left\|\eta^i K^i(k)\right\|}.$$

Therefore, the solution of (3.14) and (3.28) is stable, which leads to the SGUUB of $e^i(k)$ and $\widetilde{w}^i(k)$. □

3.2 Applications

In this section, we apply the above developed scheme (Fig. 3.1) to control a three-phase induction motor, which is one of the most used actuators for industrial applications due to its reliability, ruggedness, and relatively low cost. The control of an induction motor is challenging, since its dynamics is described by multivariable, coupled, and highly nonlinear system [13, 15]. Early works on control of induction motors was focused on the field oriented control (FOC) [7], exact input–output linearization, adaptive input–output linearization, and direct torque control (DTC) ([7] and references therein). However, most of those works were developed stabilized controllers for continuous-time model of the motor. In [13] a discrete-time model is proposed, as well as a control algorithm, assuming that the parameters and load torque of the motor model are known. Moreover, all these controllers are designed based on the physical model of the motor and results in sensitive control with respect to plant parameters variations. To this end, we consider the control problem assuming that some of the plant parameters as well as external disturbances (load torque) are unknown.

3.2.1 Motor Model

The six-order discrete-time induction motor model in the stator fixed reference frame (α, β), under the assumptions of equal mutual inductances and linear magnetic circuit, is given by [13]

$$\begin{aligned}
\omega(k+1) &= \omega(k) + \frac{\mu}{\alpha}(1-\alpha) \times M\left(i^\beta(k)\psi^\alpha(k) - i^\alpha(k)\psi^\beta(k)\right) \\
&\quad - \left(\frac{T}{J}\right) T_{\mathrm{L}}(k), \\
\psi^\alpha(k+1) &= \cos\left(n_{\mathrm{p}}\theta(k+1)\right)\rho_1(k) - \sin\left(n_{\mathrm{p}}\theta\,(k+1)\right)\rho_2(k), \\
\psi^\beta(k+1) &= \sin\left(n_{\mathrm{p}}\theta(k+1)\right)\rho_1(k) + \cos\left(n_{\mathrm{p}}\theta(k+1)\right)\rho_2(k), \\
i^\alpha(k+1) &= \varphi^\alpha(k) + \frac{T}{\sigma}u^\alpha(k) + d_1(k), \\
i^\beta(k+1) &= \varphi^\beta(k) + \frac{T}{\sigma}u^\beta(k) + d_2(k),
\end{aligned}$$

$$\theta(k+1) = \theta(k) + \omega(k)T - \frac{T_{\mathrm{L}}(k)}{J}T^2 + \frac{\mu}{\alpha}\left[T - \frac{(1-a)}{\alpha}\right] M\big(i^{\beta}(k)\psi^{\alpha}(k) - i^{\alpha}(k)\psi^{\beta}(k)\big), \quad (3.33)$$

with

$$\rho_1(k) = a\left(\cos\left(\phi(k)\right)\psi^{\alpha}(k) + \sin\left(\phi(k)\right)\psi^{\beta}(k)\right) + b\left(\cos\left(\phi(k)\right)i^{\alpha}(k) + \sin\left(\phi(k)\right)i^{\beta}(k)\right),$$

$$\rho_2(k) = a\left(\cos\left(\phi(k)\right)\psi^{\alpha}(k) - \sin\left(\phi(k)\right)\psi^{\beta}(k)\right) + b\left(\cos\left(\phi(k)\right)i^{\alpha}(k) - \sin\left(\phi(k)\right)i^{\beta}(k)\right),$$

$$\begin{aligned}\varphi^{\alpha}(k) &= i^{\alpha}(k) + \alpha\beta T\psi^{\alpha}(k) + n_{\mathrm{p}}\beta T\omega(k)\psi^{\alpha}(k) - \gamma T i^{\alpha}(k),\\ \varphi^{\beta}(k) &= i^{\beta}(k) + \alpha\beta T\psi^{\beta}(k) + n_{\mathrm{p}}\beta T\omega(k)\psi^{\beta}(k) - \gamma T i^{\beta}(k),\\ \phi(k) &= n_{\mathrm{p}}\theta(k),\end{aligned}$$

with $b = (1-a)M$, $\alpha = \frac{R_{\mathrm{r}}}{L_{\mathrm{r}}}$, $\gamma = \frac{M^2R_{\mathrm{r}}}{\sigma L_{\mathrm{r}}^2} + \frac{R_{\mathrm{s}}}{\sigma}$, $\sigma = L_{\mathrm{s}} - \frac{M^2}{L_{\mathrm{r}}}$, $\beta = \frac{M}{\sigma L_{\mathrm{r}}}$, $a = \mathrm{e}^{-\alpha T}$, $\mu = \frac{Mn_{\mathrm{p}}}{JL_{\mathrm{r}}}$, where L_{s}, L_{r}, and M are the stator, rotor, and mutual inductance, respectively; R_{s} and R_{r} are the stator and rotor resistances, respectively; n_{p} is the number of pole pairs; i^{α} and i^{β} represents the currents in the α and β phases, respectively; ψ^{α} and ψ^{β} represents the fluxes in the α and β phases, respectively; and θ is the rotor angular displacement.

3.2.2 Block-Strict-Feedback-Form (BSFF) for an Induction Motor

Let us define the following states:

$$\begin{aligned}x^1(k) &= \begin{bmatrix}\omega(k)\\ \Psi(k)\end{bmatrix}; \quad x^2(k) = \begin{bmatrix}i^{\alpha}(k)\\ i^{\beta}(k)\end{bmatrix},\\ u(k) &= \begin{bmatrix}u^{\alpha}(k)\\ u^{\beta}(k)\end{bmatrix}; \quad y_{\mathrm{d}}(k) = \begin{bmatrix}\omega_{\mathrm{d}}(k)\\ \Psi_{\mathrm{d}}(k)\end{bmatrix},\\ y(k) &= x^1(k),\end{aligned} \quad (3.34)$$

where $\Psi(k) = \psi^{\alpha^2}(k) + \psi^{\beta^2}(k)$ is the rotor flux magnitude, $\omega_{\mathrm{d}}(k)$ and $\Psi_{\mathrm{d}}(k)$ are the reference signals. The objective of control is to drive the output $y(k)$ to track the reference $y_{\mathrm{d}}(k)$. Using (3.34) the system (3.33) can be represented in the BSFF consisting of two blocks

$$\begin{aligned}x^1(k+1) &= f^1\big(x^1(k)\big) + g^1\big(x^1(k)\big)\,x^2(k) + d^1(k),\\ x^2(k+1) &= f^2\big(\overline{x}^2(k)\big) + g^2\big(\overline{x}^2(k)\big)\,u(k),\end{aligned}$$

where $f^1\big(x^1(k)\big)$, $g^1(x_1(k))$, $f_2(\overline{x}_2(k))$, and $g_2(\overline{x}_2(k))$ are assumed to be unknown and $d_1(k)$ is the unknown bounded disturbances; in this case this disturbance is the load torque. Now we use the HONN to approximate the desired virtual controls and the ideal practical controls described as

$$\begin{aligned}\alpha^{1*}(k) \triangleq x^2(k) &= \varphi^1\big(x^1(k), y_{\mathrm{d}}(k+2)\big),\\ u^*(k) &= \varphi^2\big(x^1(k), x^2(k), \alpha^{1*}(k)\big),\\ y(k) &= x^1(k).\end{aligned}$$

The HONN proposed for this application is as follows:

$$\begin{aligned}\alpha^1(k) &= {w^1}^{\top} z^1\left(\varrho^1(k)\right),\\ u(k) &= {w^2}^{\top} z^2\left(\varrho^2(k)\right),\end{aligned}$$

with

$$\begin{aligned}\varrho^1(k) &= \left[x^1(k), y_d\,(k+2)\right]^{\top},\\ \varrho^2(k) &= \left[x^1(k), x^2(k), \alpha^1(k)\right]^{\top}.\end{aligned}$$

The weights are updated using the EKF as follows:

$$\begin{aligned}w^i(k+1) &= w^i(k) + \eta^i K^i(k) e^i(k) \qquad (i = 1, 2),\\ K^i(k) &= P^i(k) H^i(k)\left[R^i(k) + {H^i}^{\top}(k) P^i(k) H^i(k)\right]^{-1},\\ P^i(k+1) &= P^i(k) - K^i(k) {H^i}^{\top}(k) P^i(k) + Q^i(k),\end{aligned}$$

with

$$\begin{aligned}e^1(k) &= y_{\mathrm{d}}(k) - y(k),\\ e^2(k) &= x^2(k) - \alpha^1(k)\,.\end{aligned}$$

The training is performed online using a series–parallel configuration. All the NN states are initialized in a random way. The associated covariances matrices are initialized as diagonals, and the nonzero elements are $P_1\,(0) = P_2\,(0) = 10000$; $Q_1\,(0) = Q_2\,(0) = 5000$, and $R_1\,(0) = R_2\,(0) = 10000$, respectively. The simulation is performing under the presence of the disturbances $d^1(k)$ as shown in Fig. 3.3 and parametric variations (Fig. 3.4).

3.2.3 Reduced Order Nonlinear Observer

The last control algorithm works with the full state measurement assumption [13]. However, the rotor fluxes measurement is a difficult task. Here, a reduced order nonlinear observer is designed for fluxes with rotor speed and current measurements only. The flux dynamics in (3.33) can be written as

$$\Psi(k+1) = aG(k)\Psi(k) + (1-a)MG(k)\mathbf{I}(k),$$

with

$$G(k) = \begin{bmatrix} \cos(n_{\mathrm{p}} T\omega(k)) & -\sin(n_{\mathrm{p}} T\omega(k)) \\ \sin(n_{\mathrm{p}} T\omega(k)) & \cos(n_{\mathrm{p}} T\omega(k)) \end{bmatrix},$$
$$\mathbf{I}(k) = \begin{bmatrix} i^{\alpha}(k) \\ i^{\beta}(k) \end{bmatrix}. \tag{3.35}$$

The proposed observer for the system (3.33) assumes the speed and current available for measurements:

$$\widehat{\Psi}(k+1) = aG(k)\widehat{\Psi}(k) + (1-a)MG(k)\mathbf{I}(k).$$

Let us define

$$e^{\Psi}(k) = \Psi(k) - \widehat{\Psi}(k).$$

Then

$$e^{\Psi}(k+1) = aG(k)e^{\Psi}(k).$$

A Lyapunov candidate function to proof stability of $e^{\Psi}(k)$ is

$$V(k) = e^{\Psi^{\top}}(k)e^{\Psi}(k), \tag{3.36}$$

with

$$\begin{aligned} \Delta V(k) &= V(k+1) - V(k) = e^{\Psi^{\top}}(k-1)\, e^{\Psi}(k+1) - e^{\Psi^{\top}}(k)e^{\Psi}(k), \\ &= e^{\Psi^{\top}}(k)\big(a^2 G^{\top}(k)G(k) - I\big)\, e^{\Psi}(k), \end{aligned}$$

where

$$a^2 G^{\top}(k)G(k) - I < 0. \tag{3.37}$$

By (3.35), $G^{\top}(k)G(k) = I$ then the condition (3.37) is reduced to

$$\begin{bmatrix} a^2 & 0 \\ 0 & a^2 \end{bmatrix} - \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} < 0,$$

where $a < 1$, $a = \mathrm{e}^{-\alpha T}$. This condition is satisfied due to the fact that T and α are always positive. So the increment of the Lyapunov function (3.36) is always negative, implying that the tracking error tends asymptotically to zero. Now we use $\widehat{\psi}^{\alpha}$ and $\widehat{\psi}^{\beta}$ to implement the control algorithm developed above.

3.2.4 Simulation Results

The simulation is performed using the system (3.33) with the parameters given in Table 3.1.

The tracking results are presented in Figs. 3.2 and 3.3. There the tracking performance can be verified for the two plant outputs. Figure 3.4 displays the load torque applied as an external disturbance. Figure 3.5 portrays a parametric variation introduced in the rotor resistance (R_{r}) as an increment. Figure 3.6 shows the weight evolution. Figures 3.7 and 3.8 display the control law signals. Figures 3.9 and 3.10 portray the fluxes and their estimates.

Table 3.1. Induction motor parameters

Parameter	Value	Description
R_{s}	$14\,\Omega$	Stator resistance
L_{s}	400 mH	Stator inductance
M	377 mH	Mutual inductance
R_{r}	$10.1\,\Omega$	Rotor resistance
L_{r}	412.8 mH	Rotor inductance
n_{p}	2	Number of pole pairs
J	$0.01\,\mathrm{Kg\,m^2}$	Moment of inertia
ω_{n}	$168.5\,\mathrm{rad\,s^{-1}}$	Nominal speed
$T_{L_{\mathrm{n}}}$	1.1 N m	Nominal load
T	0.0001 s	Sampling period

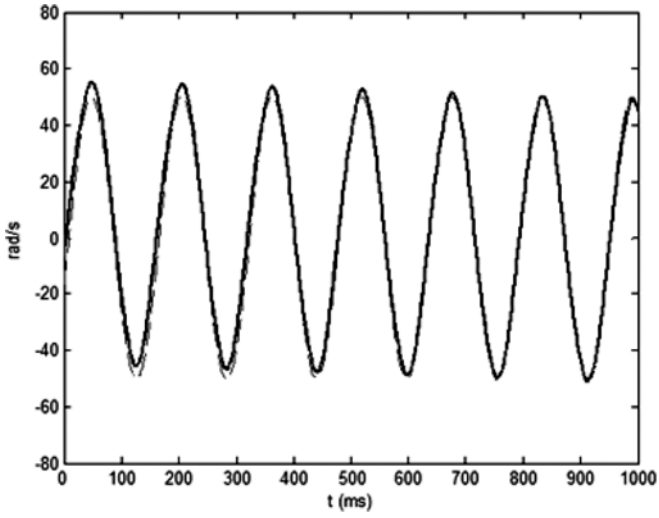

Fig. 3.2. Tracking performance $\omega(k)$ (*solid line*) and $\omega_{\mathrm{d}}(k)$ (*dashed line*)

Comment 3.3. The purpose of this chapter is to improve the tracking performance for a class of MIMO discrete-time nonlinear systems, by means of the use of the EKF as the neural network learning algorithm; this approach is validated by the simulation results presented above.

Comment 3.4. In this chapter, the causality contradiction is avoided due to the fact that the controller is constructed based on the maximum r-step ahead predictor by the backstepping technique.

Comment 3.5. In literature there are few results that present both external disturbances (load torque) and parametric changes (resistance variations) as in this book.

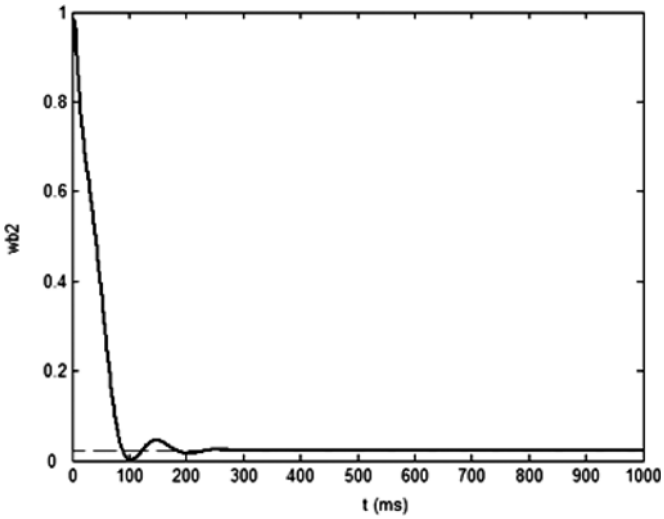

Fig. 3.3. Tracking performance $\Psi(k)$ (*solid line*) and $\Psi_{\mathrm{d}}(k)$ (*dashed line*)

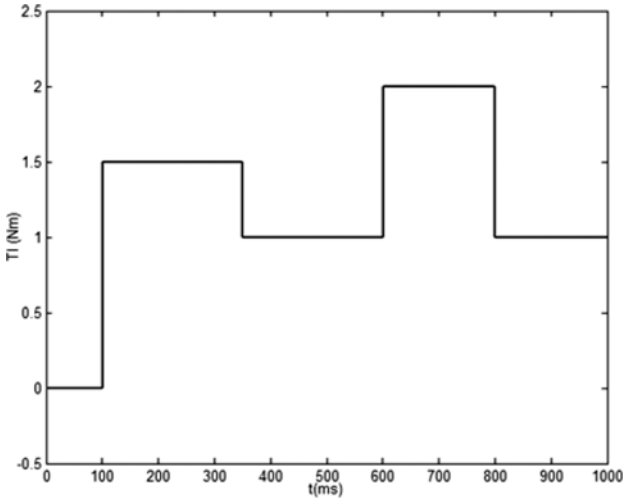

Fig. 3.4. Load torque $T_{\mathrm{L}}(k)$

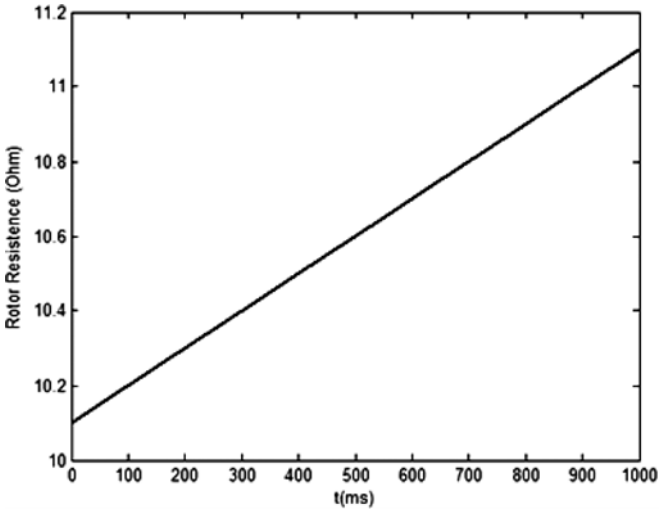

Fig. 3.5. Rotor resistance variation (R_r)

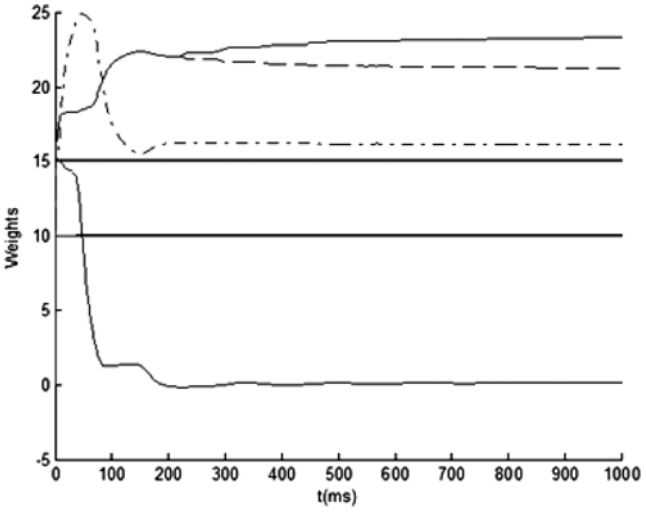

Fig. 3.6. Weights evolution

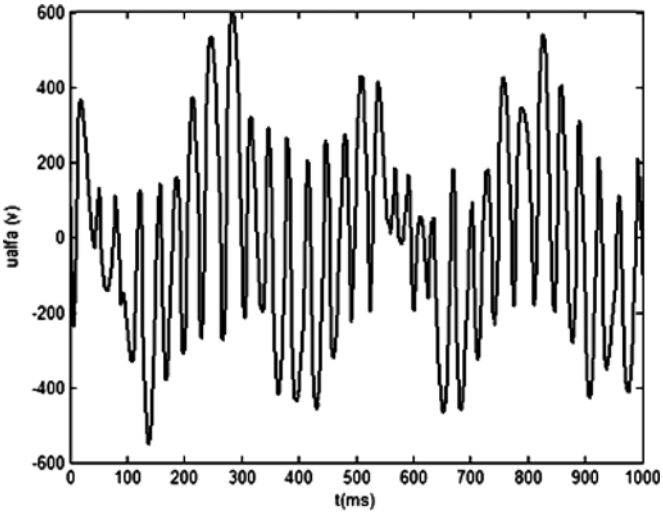

Fig. 3.7. Control law signal $u^{\alpha}(k)$

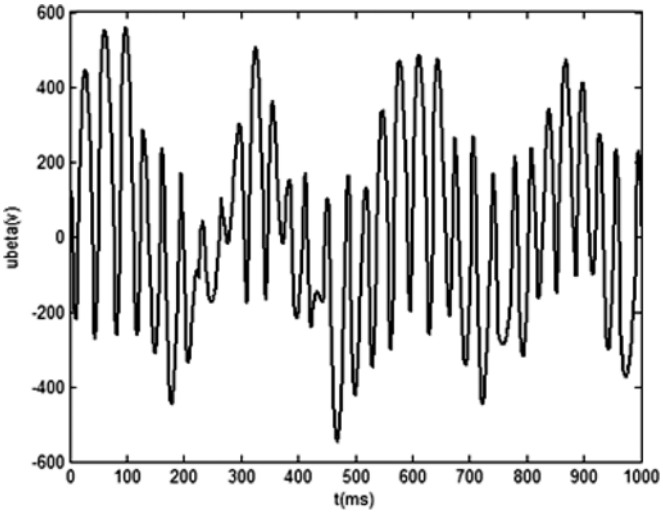

Fig. 3.8. Control law signal $u^{\beta}(k)$

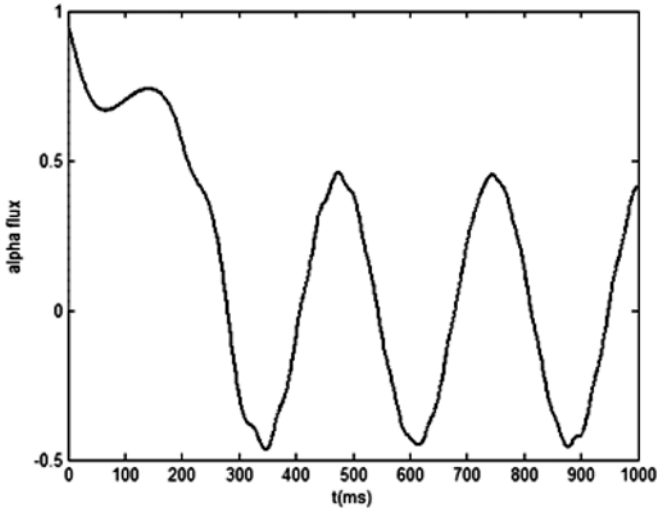

Fig. 3.9. Time evolution of $\psi^{\alpha}(k)$ and its estimate (real in *solid line* and estimated in *dashed line*)

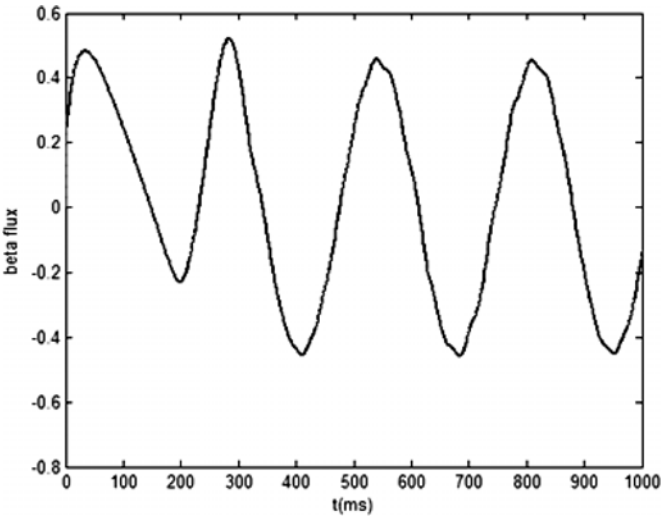

Fig. 3.10. Time evolution of $\psi^{\beta}(k)$ and its estimate (real in *solid line* and estimated in *dashed line*)

3.3 Conclusions

This chapter has presented the application of HONN to solve the tracking problem for a class of MIMO discrete-time nonlinear systems, using the backstepping technique. The training of the neural network is performed online using an extended Kalman filter. The boundness of the tracking error is established on the basis of the Lyapunov approach. The HONN training with the learning algorithm based in EKF presents good performance even in presence of larger bounded disturbances such as load torque variations and change on the plant parameters (resistance change). Based on the proposed control scheme, a robust neural controller is designed for an induction motor.