7

Real Time Implementation

In this chapter real time implementation is presented in order to validate the theoretical results discussed in previous chapters. The results presented in this chapter include the Neural Network Identification scheme presented in Chap. 4, the RHONO presented in Chap. 5, the Neural Backstepping Approach analyzed in Chap. 3, the Neural Bock Control Technique discussed in Chap. 4 and the modifications of the last two controllers treated in Chap. 6 to include the RHONO. All these applications was performed using a three phase induction motor.

The experiments are performed using a benchmark, which includes a PC for supervising, a PWM unit for the power stage, a dSPACE DS1104 board for data acquisition and control of the system (dSPACE is a trademark of dSPACE GmbH), and a three phase induction motor as the plant has to be controlled, with the following characteristics: 220 V, 60 Hz, 0.19 kW, 1,660 rpm, 1.3 A [1]. Series of photographs and figures of the benchmark are included. Figure 7.1 presents a schematic representation of the benchmark used in these experiments. Figure 7.2 displays the encoder coupled with an induction motor, Fig. 7.3 presents a view of the PC and the DS1104 board, and Fig. 7.4 shows the PWM driver. The DS1104 board allows to download applications directly from Simulink (Matlab and Simulink are trademarks of the MathWorks Inc.) as is shown in Fig. 7.5. In Fig. 7.6, a Desktop interface for the DS1104 board is included in order to clarify the visualization of the experiments. The experiments performed in the benchmark includes the Neural Network Identification scheme presented in Chap. 4, the RHONO presented in Chap. 5, the Neural Backstepping Approach analyzed in Chap. 3, the Neural Bock Control Technique discussed in Chap. 4, and the modifications of the last two controllers treated in Chap. 6 to include the RHONO. Finally, the mentioned experiments are tested with a constant load torque applied with a DC generator coupled to an induction motor as shown in Figs. 7.7 and 7.8.

E.N. Sanchez et al.: *Discrete-Time High Order Neural Control – Trained with Kalman Filtering*, Studies in Computational Intelligence (SCI) **112**, 73–94 (2008)
www.springerlink.com

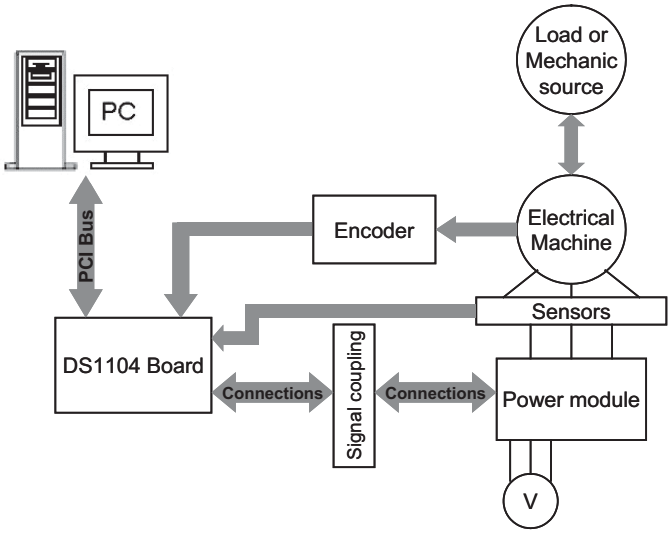

Fig. 7.1. Schematic representation of the control prototype

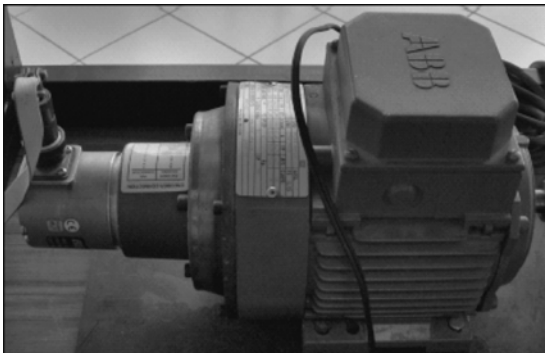

Fig. 7.2. Encoder coupled with the induction motor

7.1 Neural Identification

In this section the neural network identification scheme proposed in Chap. 4 for the discrete-time induction motor model is applied in real time to the benchmark described above. During the identification process the plant and the NN operates in open-loop. Both of them (plant and NN) have the same input vector $\left[\, u_\alpha \; u_\beta \,\right]^\top$; u_α and u_β are chirp functions with 200 V of amplitude and incremental frequencies from 0 to 150 Hz and 0 to 200 Hz, respectively. The implementation is performed with a sampling time of 0.0005 s. The results of the real-time implementation are presented as follows: Fig. 7.9 shows the identification of rotor angular displacement; Fig. 7.10 displays the identification

Fig. 7.3. View of the PC and the DS1104 board

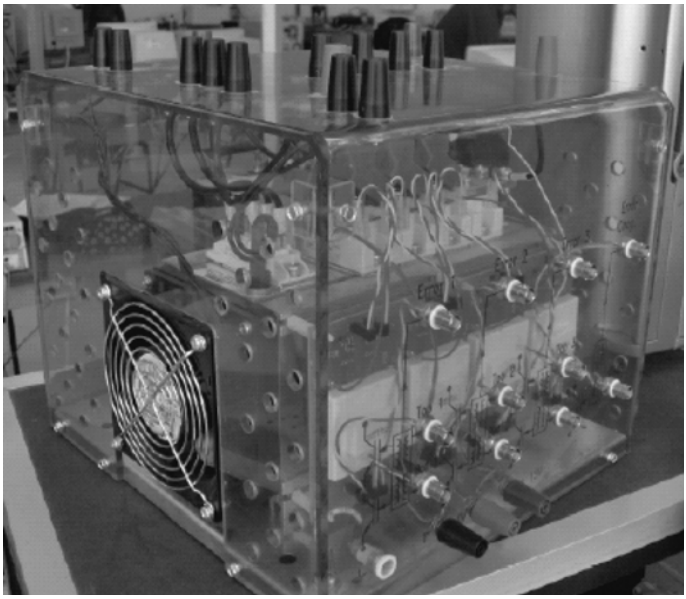

Fig. 7.4. PWM driver

performance for the speed rotor; Figs. 7.11 and 7.12 present the identification performance for the fluxes in phase α and β, respectively. Figures 7.13 and 7.14 portray the identification performance for currents in phase α and β, respectively. Finally, the input signals are presented in Fig. 7.15.

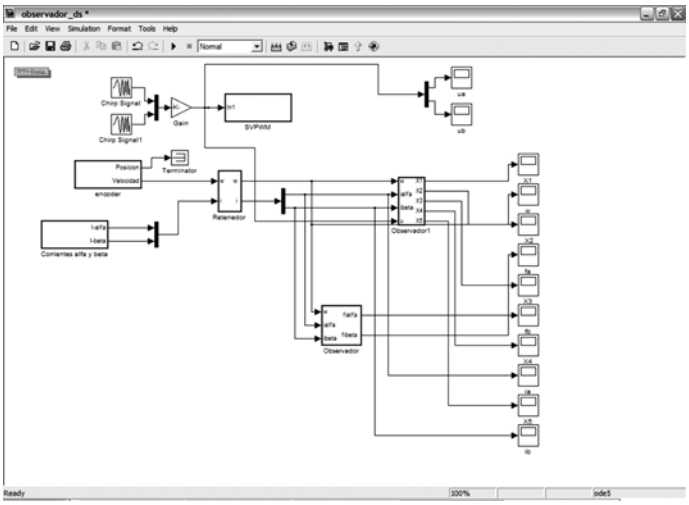

Fig. 7.5. Simulink program to be downloaded to the DS1104 board directly

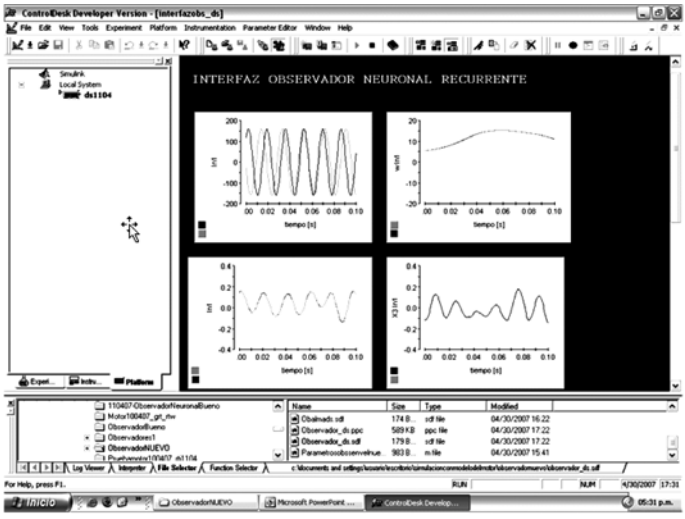

Fig. 7.6. Desktop interface for the DS1104 board

7.2 Neural State Estimation

This section presents the neural network observer (RHONO) scheme proposed in Chap. 5 for the discrete-time induction motor model as applied in real time to the benchmark described above. During the estimation process, the plant and the NN operates in open-loop. Both of them (plant and NN) have the same input vector $\left[u_\alpha \; u_\beta\right]^\top$; u_α and u_β are chirp functions with

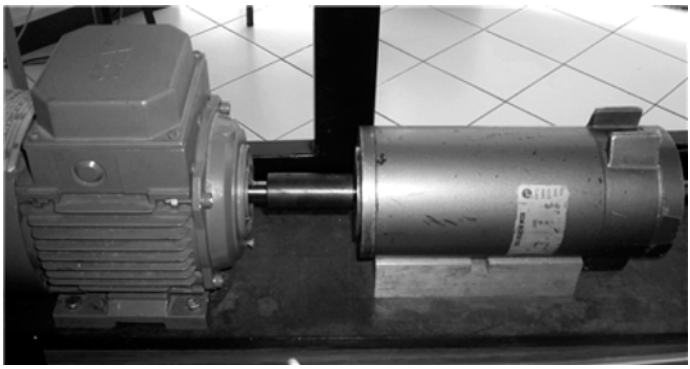

Fig. 7.7. DC generator coupled to the induction motor as a constant load torque

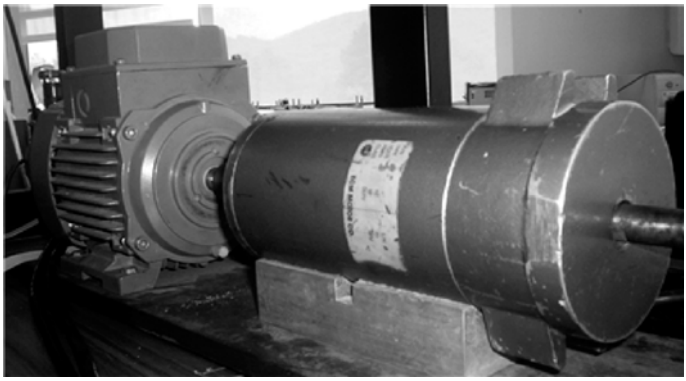

Fig. 7.8. Complete view of the DC generator coupled to the induction motor as a constant load torque

200 V of amplitude and incremental frequencies from 0 to 150 Hz and 0 to 200 Hz, respectively. The implementation is performed with a sampling time of 0.0005 s. The results of the real-time implementation are presented as follows: Fig. 7.16 displays the estimation performance for the speed rotor; Figs. 7.17 and 7.18 present the estimation performance for the fluxes in phase α and β, respectively. Figures 7.19 and 7.20 portray the estimation performance for currents in phase α and β, respectively. Finally, the input signals are presented in Fig. 7.21.

7.3 Neural Backstepping Control

This section describes the real time results of the control law designed in Chap. 3, based on the backstepping technique approximated by a HONN, for the discrete-time induction motor model with a sampling time of 0.0005 s as

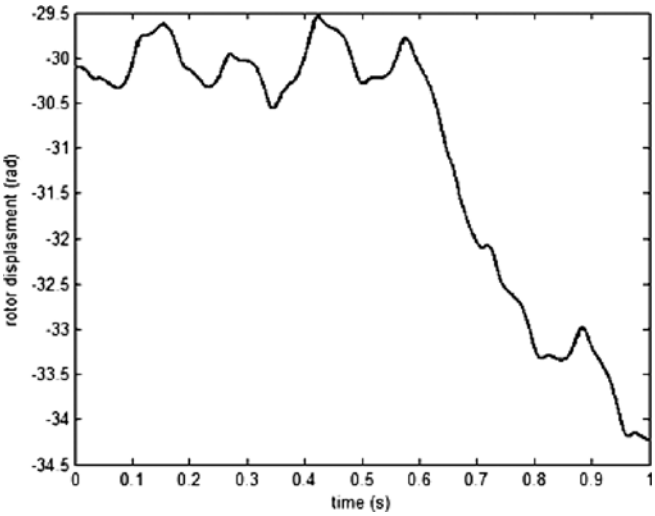

Fig. 7.9. Real time rotor displacement identification (plant signal in *solid line* and neural signal in *dashed line*)

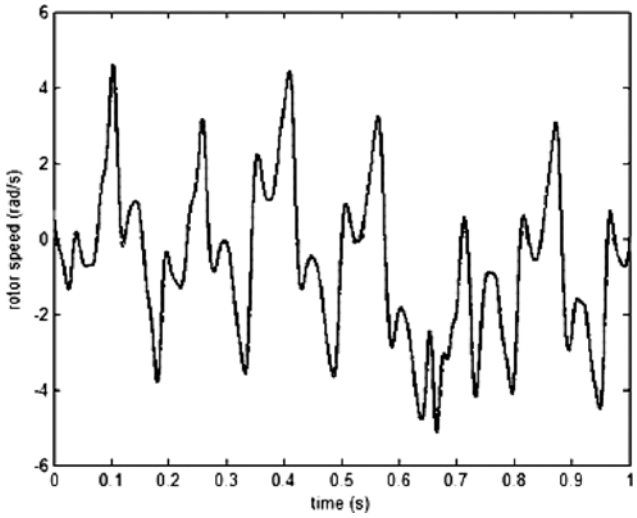

Fig. 7.10. Real time rotor speed identification (plant signal in *solid line* and neural signal in *dashed line*)

follows: The tracking results for the rotor speed and for the flux magnitude are presented in Figs. 7.22 and 7.23 for the induction motor working without load, respectively; Fig. 7.24 shows the control law in phases α and β; Fig. 7.25 presents the tracking result for the rotor speed under the presence of a load. Finally, Fig. 7.26 displays the tracking result under the presence of an external disturbance.

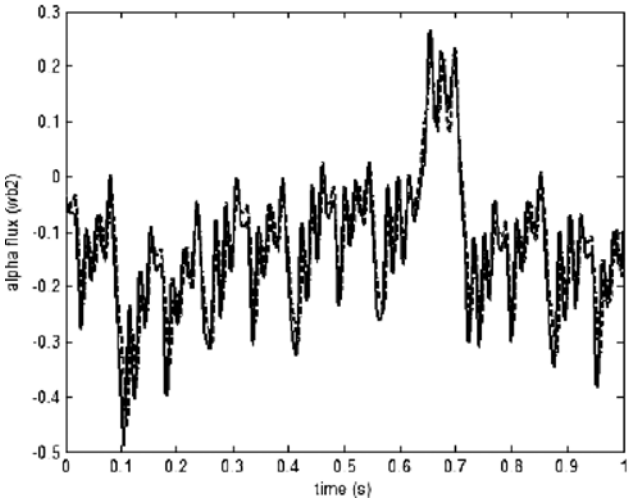

Fig. 7.11. Real time alpha flux identification (plant signal in *solid line* and neural signal in *dashed line*)

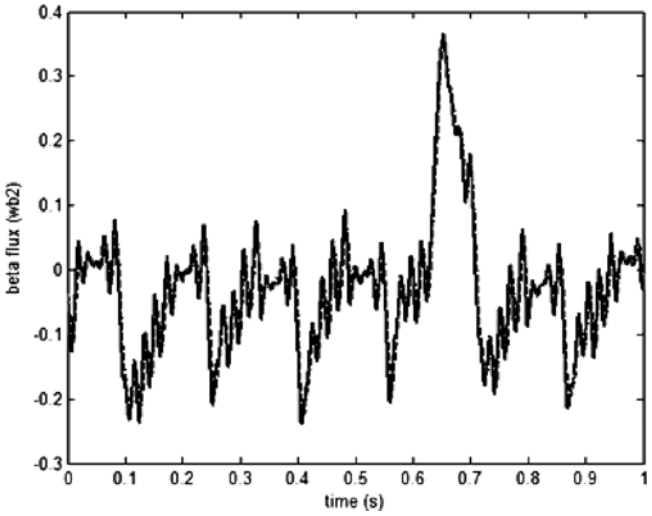

Fig. 7.12. Real time rotor beta flux identification (plant signal in *solid line* and neural signal in *dashed line*)

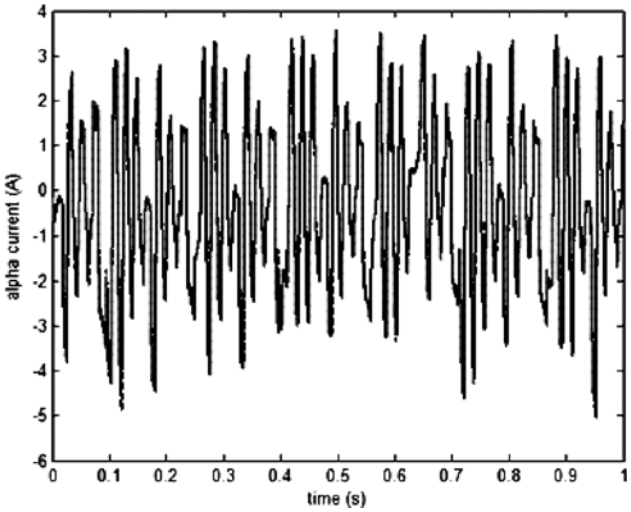

Fig. 7.13. Real time rotor alpha current identification (plant signal in *solid line* and neural signal in *dashed line*)

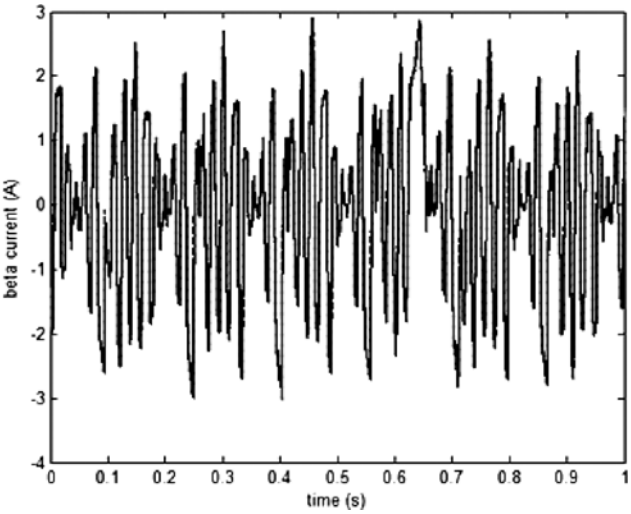

Fig. 7.14. Real time beta current speed identification (plant signal in *solid line* and neural signal in *dashed line*)

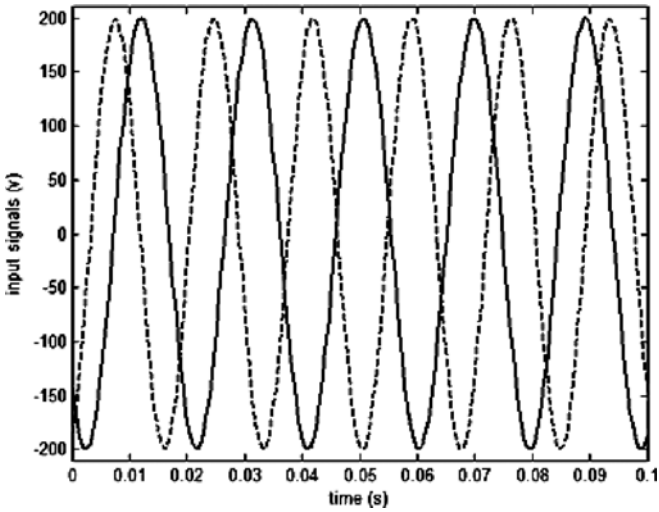

Fig. 7.15. Input signals applied during the identification process ($u^{\alpha}(k)$ in *solid line* and $u^{\beta}(k)$ in *dashed line*)

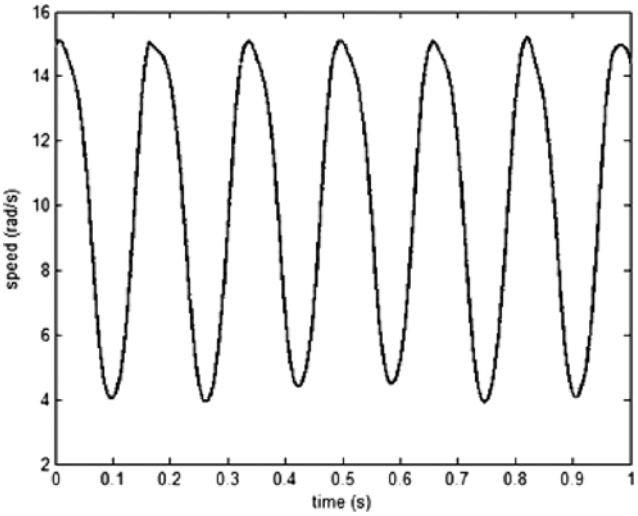

Fig. 7.16. Real time rotor speed estimation (plant signal in *solid line* and neural signal in *dashed line*)

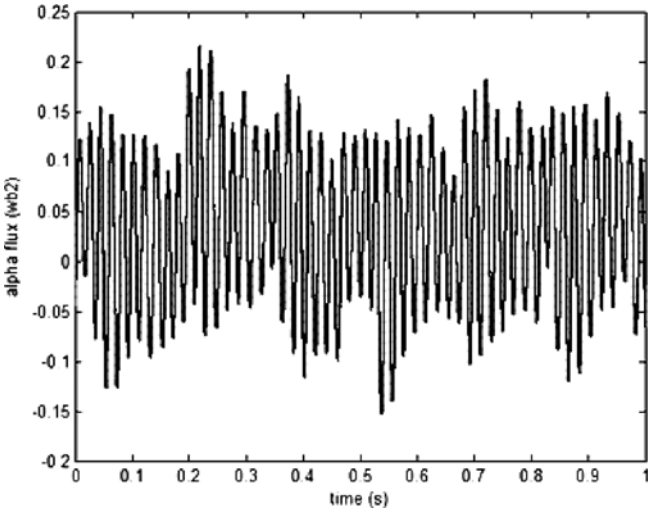

Fig. 7.17. Real time alpha flux estimation (plant signal in *solid line* and neural signal in *dashed line*)

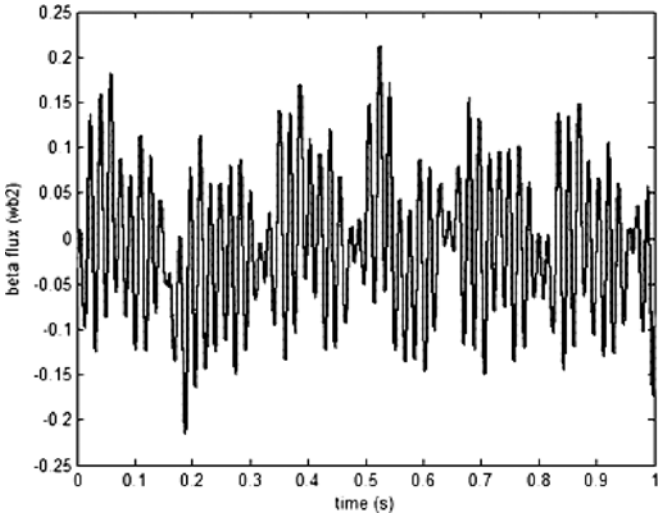

Fig. 7.18. Real time beta flux estimation (plant signal in *solid line* and neural signal in *dashed line*)

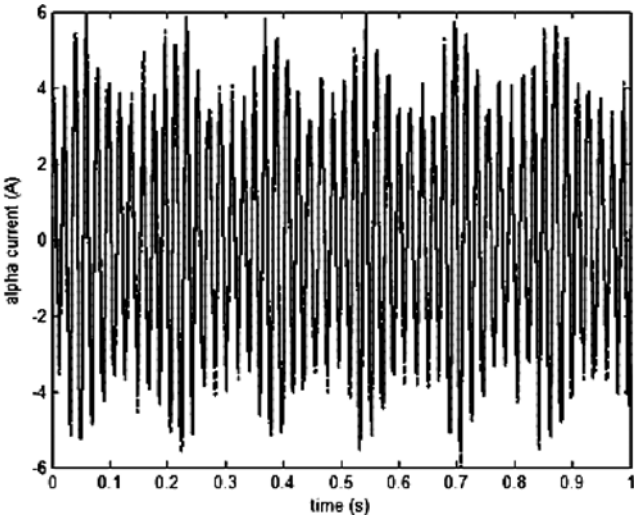

Fig. 7.19. Real time alpha current estimation (plant signal in *solid line* and neural signal in *dashed line*)

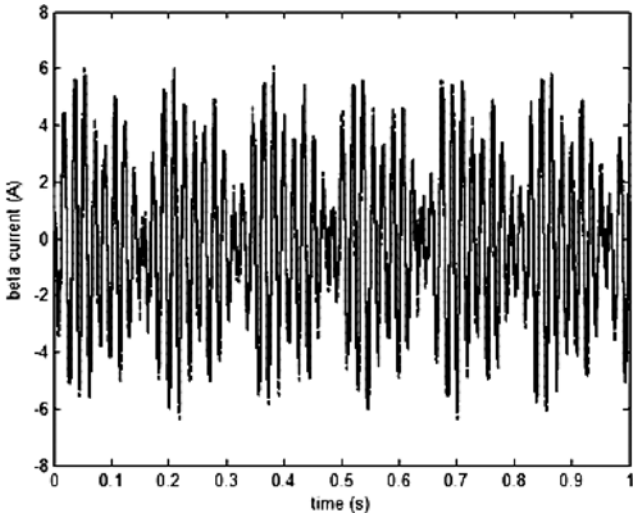

Fig. 7.20. Real time beta current estimation (plant signal in *solid line* and neural signal in *dashed line*)

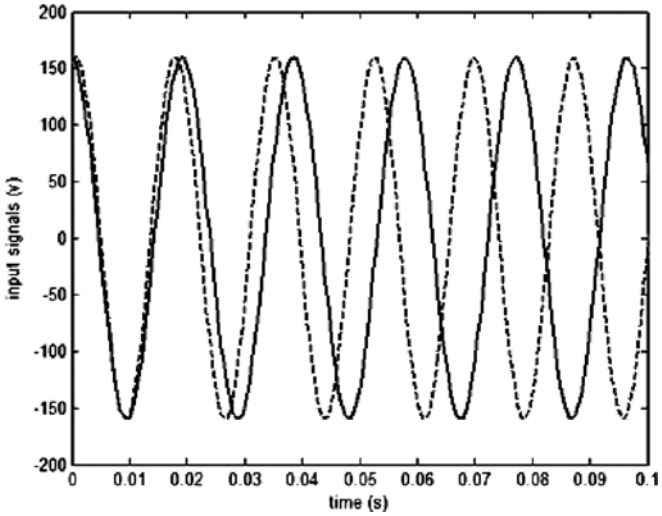

Fig. 7.21. Input signals applied during the state estimation process ($u^{\alpha}(k)$ in *solid line* and $u^{\beta}(k)$ in *dashed line*)

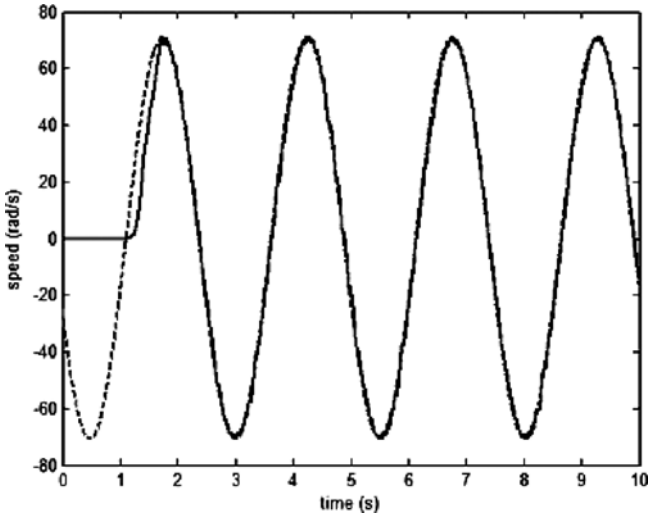

Fig. 7.22. Speed tracking performance (plant signal in *solid line* and reference signal in *dashed line*)

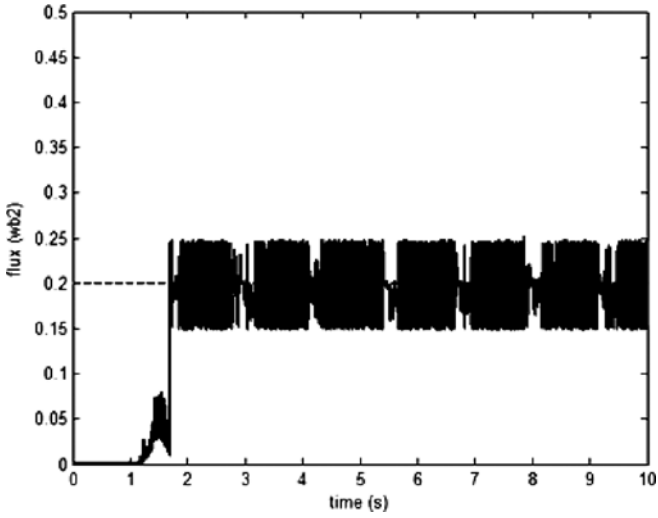

Fig. 7.23. Flux magnitude tracking performance (plant signal in *solid line* and reference signal in *dashed line*)

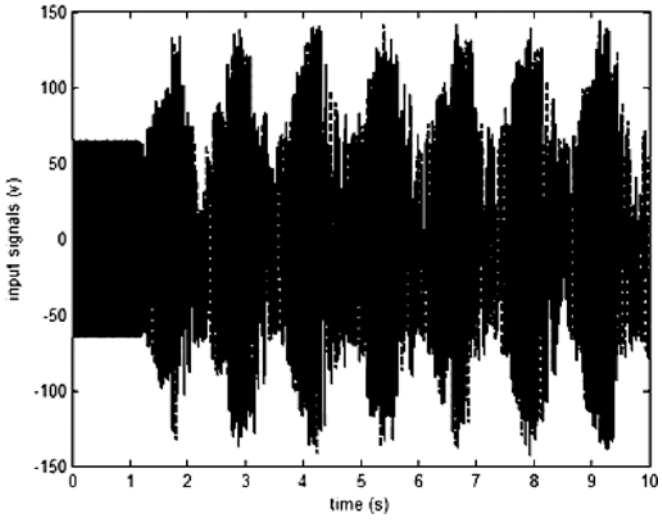

Fig. 7.24. Control law signals $u^{\alpha}(k)$ (*solid line*) and $u^{\beta}(k)$ (*dashed line*)

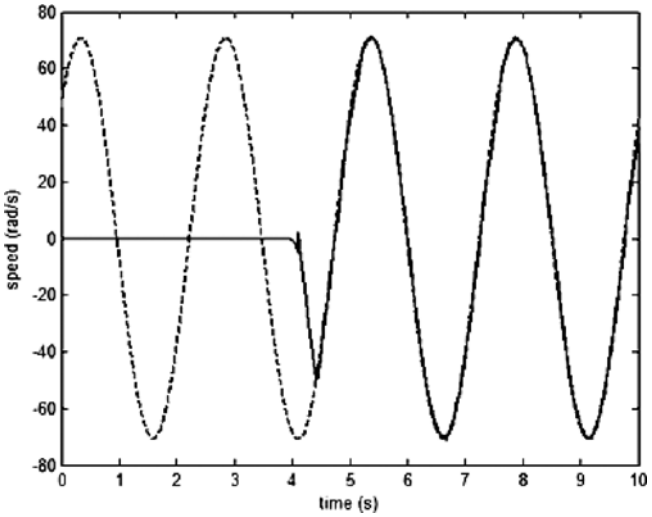

Fig. 7.25. Speed tracking performance (plant signal in *solid line* and reference signal in *dashed line*) with a constant load torque

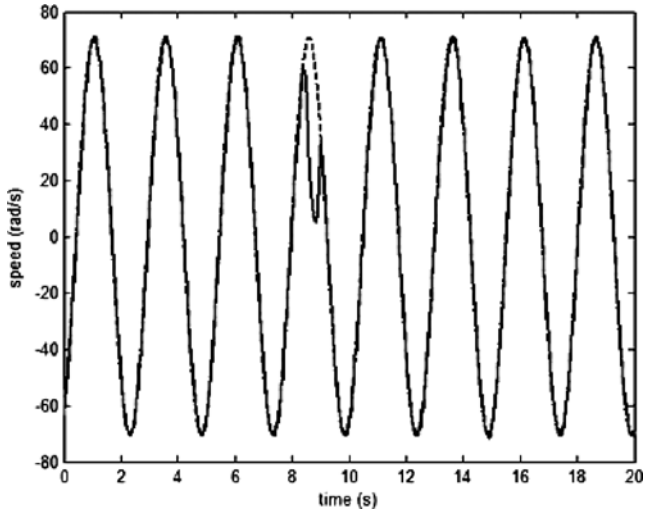

Fig. 7.26. Speed tracking performance (plant signal in *solid line* and reference signal in *dashed line*) under the presence of disturbances

7.4 Backstepping Control Using an RHONO

The real time results of the control law designed in Chap. 6, based on the backstepping technique approximated by a HONN using an RHONO, for the discrete-time induction motor model with a sampling time of 0.001 s are

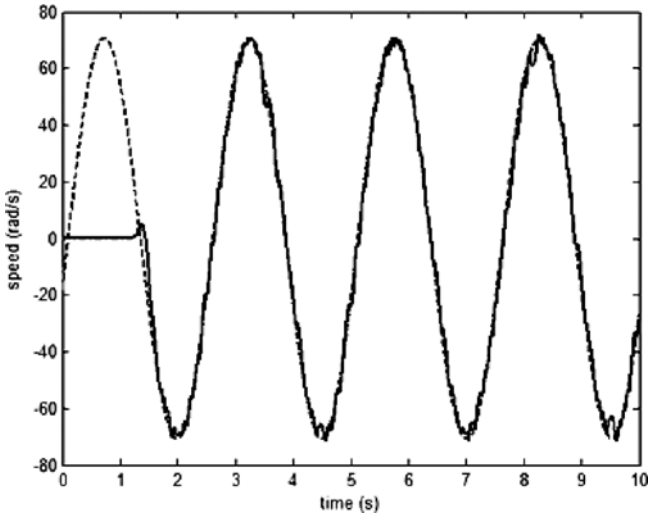

Fig. 7.27. Speed tracking performance (plant signal in *solid line*, neural signal in *dash-dot line*, and reference signal in *dashed line*)

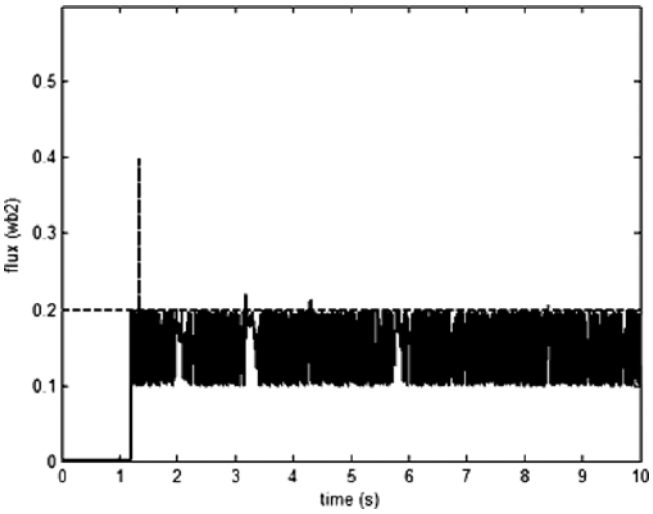

Fig. 7.28. Flux magnitude tracking performance (plant signal in *solid line*, neural signal in *dash-dot line*, and reference signal in *dashed line*)

presented as follows: The tracking results for the rotor speed and for the flux magnitude are presented in Figs. 7.27 and 7.28 for the induction motor working without load, respectively; Fig. 7.29 shows the control law in phases α and β; Fig. 7.30 presents the tracking result for the rotor speed under the presence of a load. Finally, Fig. 7.31 displays the tracking result under the presence of an external disturbance.

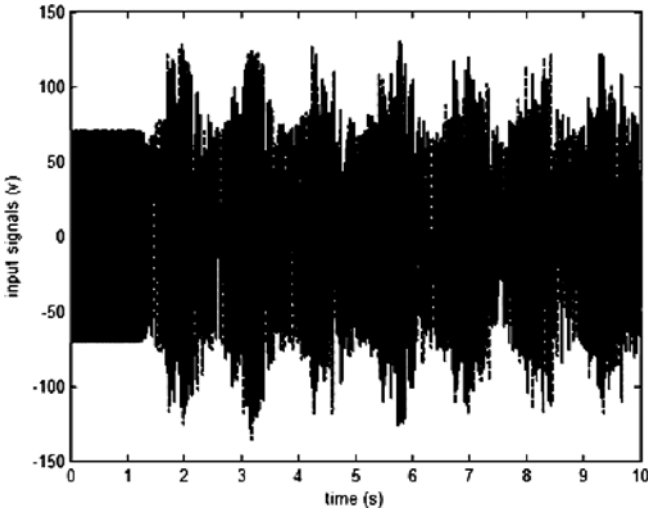

Fig. 7.29. Control law signals $u^{\alpha}(k)$ (*solid line*) and $u^{\beta}(k)$ (*dashed line*)

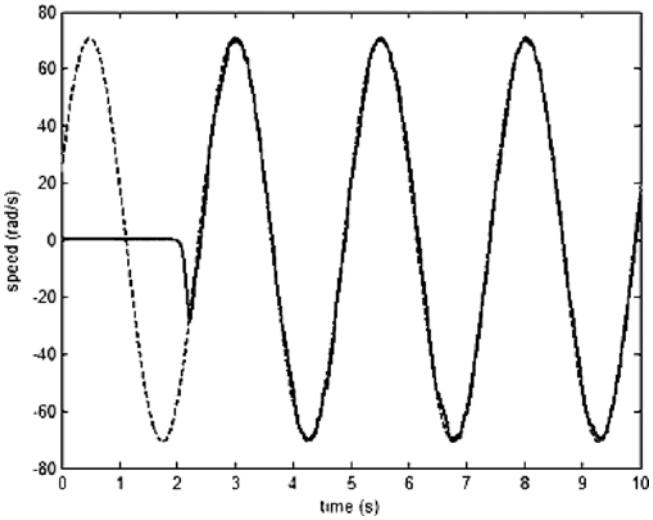

Fig. 7.30. Speed tracking performance (plant signal in *solid line*, neural signal in *dash-dot line*, and reference signal in *dashed line*) with a constant load torque

7.5 Neural Block Control with Sliding Modes

The corresponding real time results of the control law designed in Chap. 4, based on the block control and sliding modes techniques, for the discrete-time induction motor with a sampling time of 0.001 s are presented as follows: The tracking results for the rotor speed and for the flux magnitude are presented

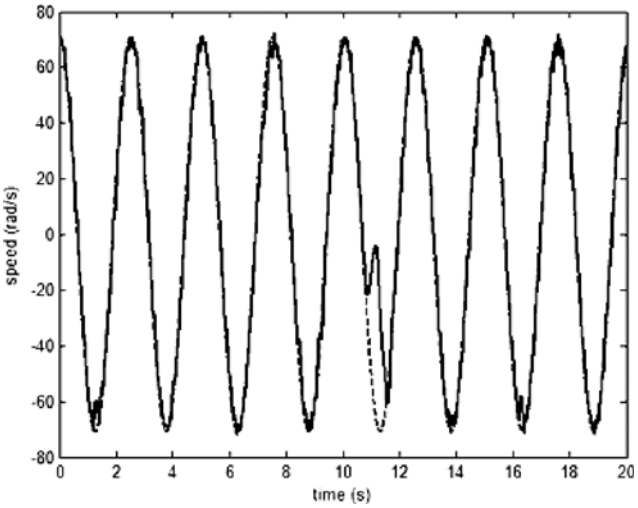

Fig. 7.31. Speed tracking performance (plant signal in *solid line*, neural signal in *dash-dot line*, and reference signal in *dashed line*) under the presence of disturbances

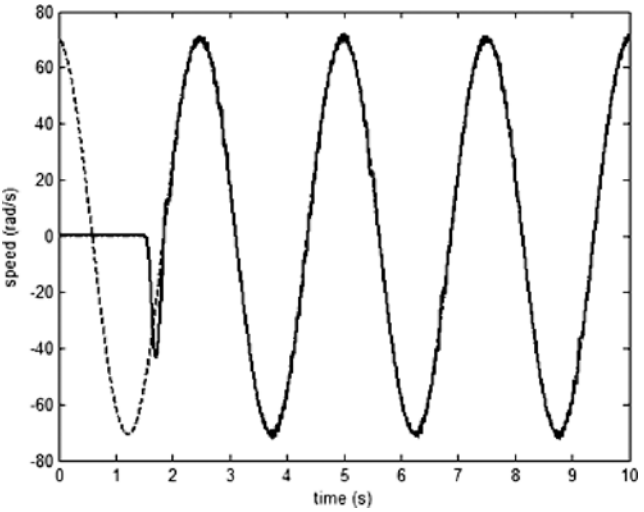

Fig. 7.32. Speed tracking performance (plant signal in *solid line*, neural signal in *dash-dot line*, and reference signal in *dashed line*)

in Figs. 7.32 and 7.33 for the induction motor working without load, respectively; Fig. 7.34 shows the control law in phases α and β; Fig. 7.35 presents the tracking result for the rotor speed with a constant load. Finally, Fig. 7.36 displays the tracking result under the presence of an external disturbance.

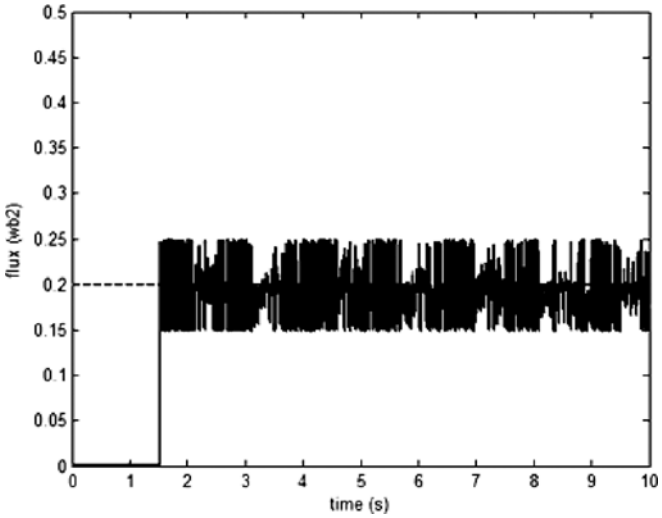

Fig. 7.33. Flux magnitude tracking performance (plant signal in *solid line*, neural signal in *dash-dot line*, and reference signal in *dashed line*)

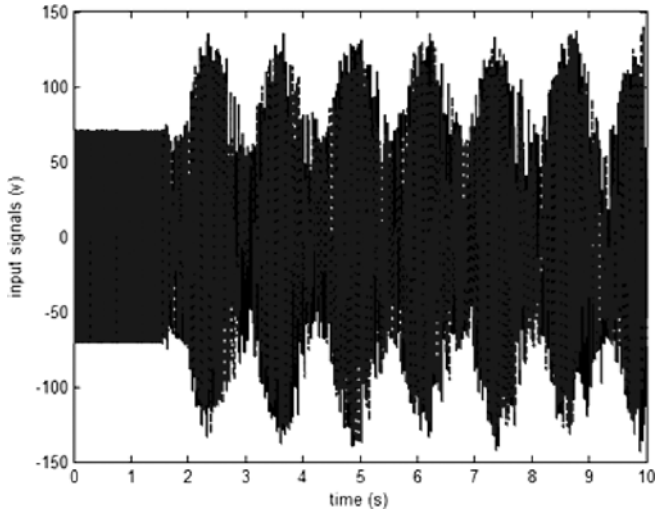

Fig. 7.34. Control law signals $u^{\alpha}(k)$ (*solid line*) and $u^{\beta}(k)$ (*dashed line*)

7.6 Block Control with Sliding Modes Using an RHONO

This section presents the real time results of the control law designed in Chap. 6 using the block control and sliding modes techniques, based on an RHONO model, for the discrete-time induction motor model with a sampling time of 0.001 s as follows: The tracking results for the rotor speed and for the

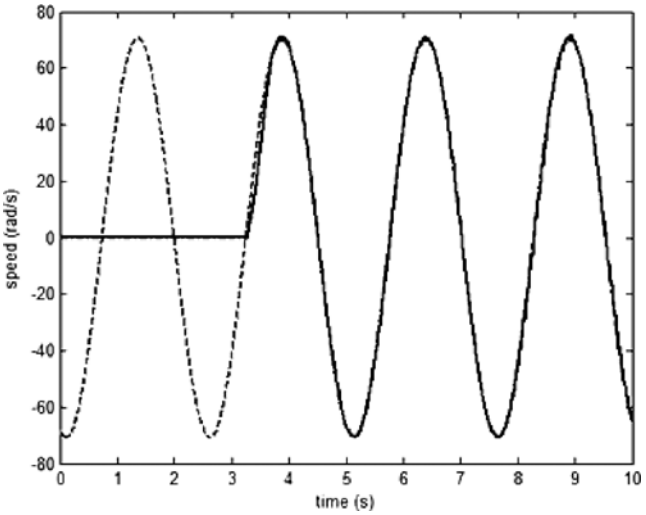

Fig. 7.35. Speed tracking performance (plant signal in *solid line*, neural signal in *dash-dot line*, and reference signal in *dashed line*) with constant load torque

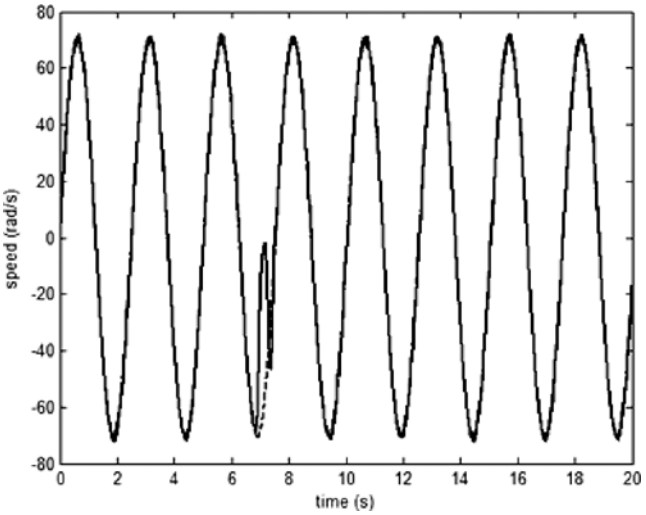

Fig. 7.36. Speed tracking performance (plant signal in *solid line*, neural signal in *dash-dot line*, and reference signal in *dashed line*) under the presence of disturbances

flux magnitude are presented in Figs. 7.37 and 7.38 for the induction motor working without load, respectively; Fig. 7.39 shows the control law in phases α and β; Fig. 7.40 presents the tracking result for the rotor speed under the presence of a load. Finally, Fig. 7.41 displays the tracking result under the presence of an external disturbance.

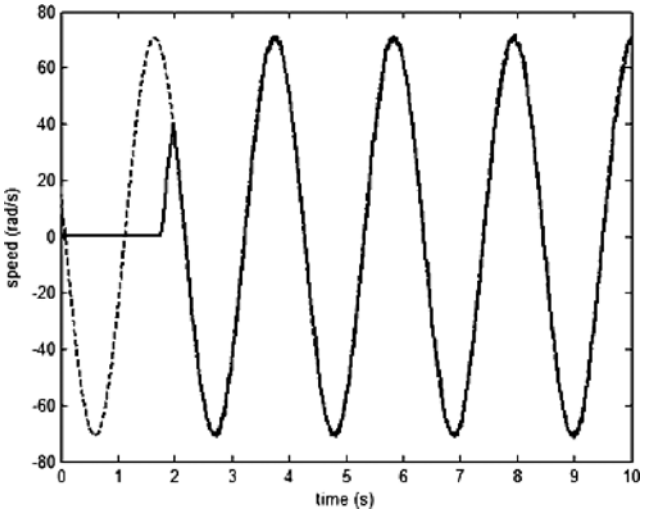

Fig. 7.37. Speed tracking performance (plant signal in *solid line*, neural signal in *dash-dot line*, and reference signal in *dashed line*)

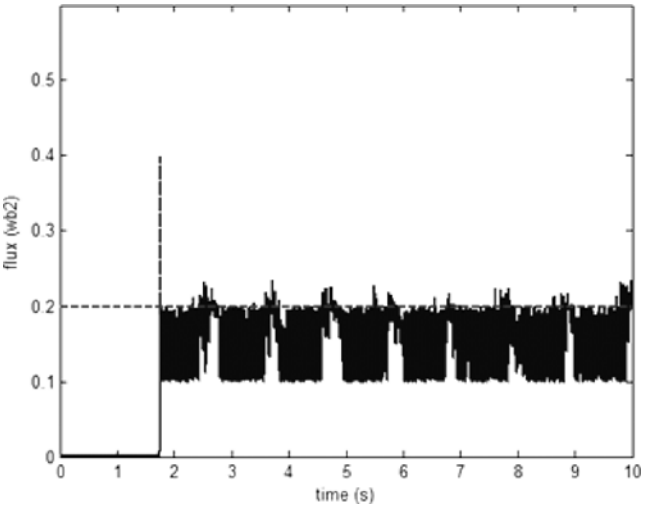

Fig. 7.38. Flux magnitude tracking performance (plant signal in *solid line*, neural signal in *dash-dot line*, and reference signal in *dashed line*)

7.7 Conclusions

To end this chapter, a comparative analysis of the four proposed schemes of control is included. In Table 7.1, the four schemes are compared with an induction motor operating without load and Table 7.2 establishes the comparison

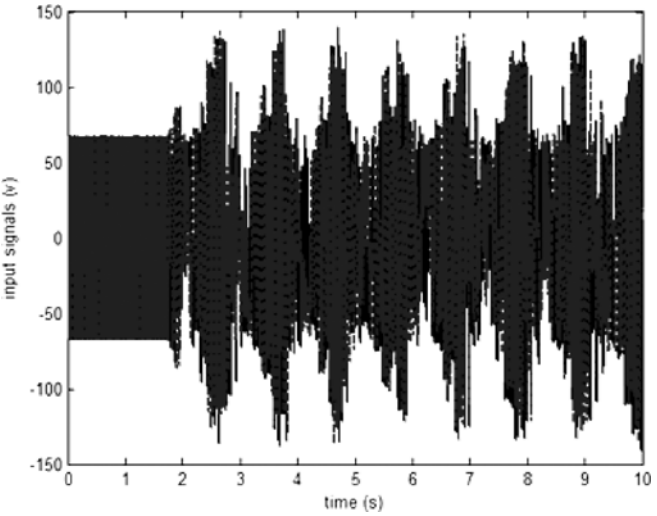

Fig. 7.39. Control law signals $u^{\alpha}(k)$ (*solid line*) and $u^{\beta}(k)$ (*dashed line*)

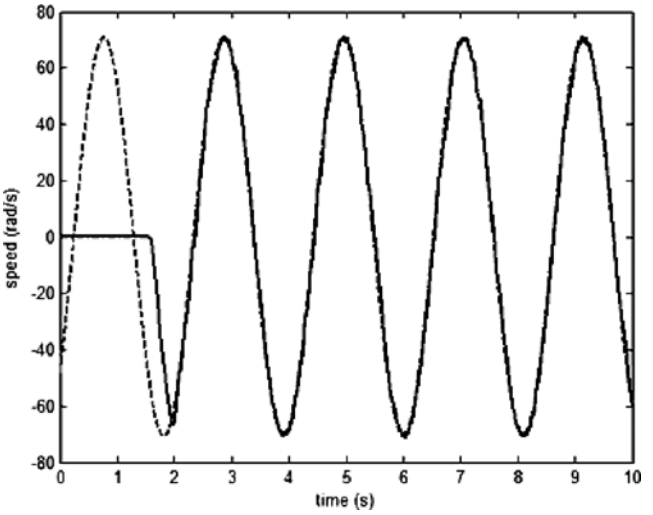

Fig. 7.40. Speed tracking performance (plant signal in *solid line*, neural signal in *dash-dot line*, and reference signal in *dashed line*) with constant load torque

between the four schemes with the motor operating in presence of a constant load.

For Tables 7.1 and 7.2, B means backstepping technique, BNO means backstepping technique using an RHONO, BCNI means block control and sliding modes techniques, and BCNO means block control and sliding modes techniques using an RHONO.

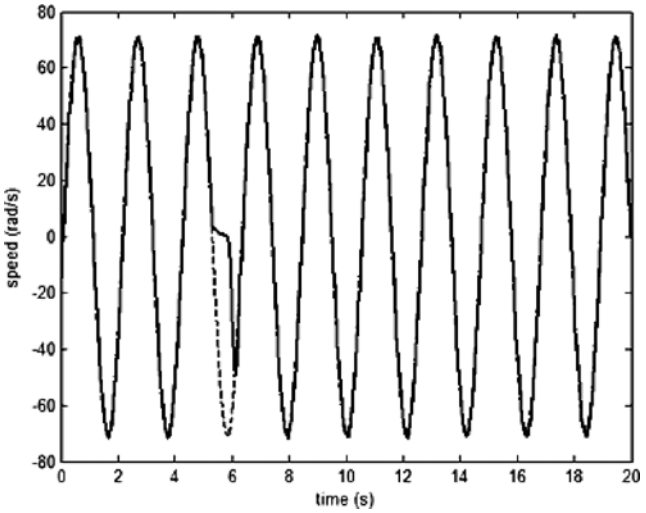

Fig. 7.41. Speed tracking performance (plant signal in *solid line*, neural signal in *dash-dot line*, and reference signal in *dashed line*) under the presence of disturbances

Table 7.1. Comparison of the mean square error for the controllers without load

Control algorithm	Mean square error
B	4.0130
BNO	8.4350
BCNI	1.9504
BCNO	4.5363

Table 7.2. Comparison of the mean square error for the controllers with load

Control algorithm	Mean square error
B	3.9158
BNO	4.6576
BCNI	1.7160
BCNO	5.2302

According to the mean square error presented above, the scheme with better performance are the ones based on the block control using the neural identifier and backstepping techniques, on the other hand the scheme with worse performance is the one based on the backstepping technique using the neural observer. It is important to remark that all the schemes present an excellent performance. However, the technique with the smaller computational complexity is the one based on the backstepping technique, since it allows to use a sampling time of 0.5 ms, whereas the other three schemes requires 1 ms.