CINVESTAV

Centro de Investigación y de Estudios Avanzados del I.P.N.
Unidad Guadalajara

Control de un Motor de Corriente Directa basado en Redes Neuronales Recurrentes

Tesis que presenta:

Carlos Eduardo Castañeda Hernández

para obtener el grado de:

Doctor en Ciencias

en la especialidad de:

Ingeniería Eléctrica

Directores de tesis:

Dr. Edgar Nelson Sánchez Camperos

Dr. Bernardino Castillo-Toledo

Guadalajara, Jalisco, Agosto de 2009.

ID: 163420-1001

CINVESTAV

Centro de Investigación y de Estudios Avanzados del I.P.N.
Unidad Guadalajara

DC Motor Control based on Recurrent Neural Networks

A thesis presented by:

Carlos Eduardo Castañeda Hernández

to obtain the degree of:

Doctor in Science

in the subject of:

Electrical Engineering

Thesis Advisors:

Dr. Edgar Nelson Sánchez Camperos

Dr. Bernardino Castillo-Toledo

Guadalajara, Jalisco, August 2009.

Control de un Motor de Corriente Directa basado en Redes Neuronales Recurrentes

Tesis de Doctorado en Ciencias Ingeniería Eléctrica

Por:

Carlos Eduardo Castañeda Hernández

Maestro en Ingeniería de Sistemas Electrónicos y Computacionales

Universidad De La Salle Bajío 2001-2003

Becario de CONACYT, expediente no. 203575

Directores de Tesis:

Dr. Edgar Nelson Sánchez Camperos

Dr. Bernardino Castillo-Toledo

CINVESTAV del IPN Unidad Guadalajara, Agosto de 2009.

DC Motor Control based on Recurrent Neural Networks

Doctor of Science Thesis
In Electrical Engineering

By:

Carlos Eduardo Castañeda Hernández

Master in Engineering of Electronics and Computer Systems

Universidad De La Salle Bajío 2001-2003

Scholarship granted by CONACYT, no. 203575

Thesis Advisors:

Dr Edgar Nelson Sánchez Camperos

Dr Bernardino Castillo-Toledo

CINVESTAV del IPN Unidad Guadalajara, August 2009.

Resumen

Las máquinas eléctricas de corriente directa son versátiles y muy utilizadas en nuestra vida diaria [39]. Estas máquinas pueden ser usadas como motores y se utilizan en el transporte, en la industria y en el hogar, y en muchas otras aplicaciones. Este es el caso del motor de corriente directa con excitación en el devanado de campo, el cual es utilizado en esta tesis.

Por otro lado, se conoce que las redes neuronales se utilizan en el diseño de sistemas de identificación y control de sistemas no lineales complejos; esto es debido a que presentan buena capacidad de aprendizaje y adaptación ante la presencia de perturbaciones externas y errores en el modelado.

Dentro de los algoritmos de entrenamiento que existen en la actualidad para redes neuronales, presentan algunas desventajas como: lento aprendizaje, alta sensibilidad a las condiciones iniciales, mínimos locales, entre otras. Como una alternativa viable se han propuesto algoritmos de entrenamiento de redes neuronales basados en el filtro de Kalman ([12], [13], [41]).

Por otra parte, es sabido que la técnica de control por modos deslizantes es robusta ante la presencia de perturbaciones externas que se encuentran en el subespacio de control [8]. Esta técnica es una herramienta efectiva en el control de sistemas no lineales; sin embargo, en tiempo continuo, su alta frecuencia de conmutación resulta ser una desventaja. Aunque los modos deslizantes en tiempo discreto no presentan este problema.

Esta tesis presenta la solución de un problema de seguimiento de trayectoria de velocidad angular y par electromagnético de un motor de corriente directa con excitación en el devanado de campo. Esto se logra mediante la identificación del modelo de la planta utilizando una red neuronal recurrente de alto orden en tiempo discreto. Basado en el modelo de identificación neuronal, se aplican las técnicas de control por bloques y modos deslizantes discretos. Además se muestra el análisis de estabilidad del sistema completo que se realiza bajo el esquema de Lyapunov. Para validar el análisis teórico, los resultados se presentan mediante la simulación y finalmente se diseña y construye un prototipo, que es utilizado para la implementación en tiempo real de los esquemas de control propuestos.

Abstract

Direct current (DC) machines are versatile and extensively used every day [39]. This kind of machines can be used as motor and their applications can be in transportation, industry, household, among others. This is the case of the DC motor with separate winding excitation, which is presented in this dissertation.

In the other hand, it is known that neural networks are used in identification and control of nonlinear complex systems, due to its good capacity of learning and adaptation in presence of external disturbances and uncertainties.

Usually, training algorithms for neural networks present disadvantages such as slow learning speed, local minima and high sensitivity to initial conditions, among others. Hence, as a very viable alternative, training algorithms for neural networks based on Kalman filtering have been recently proposed ([12], [13], [41]).

Also, it is known that the sliding mode control technique achieves robustness with respect to matched perturbations [8]. This technique is an effective tool for controlling nonlinear systems; nevertheless, in continuous-time, its high frequency of switching is a disadvantage. Although, the sliding modes in discrete-time do not present this problem.

This dissertation presents the solution of trajectory tracking of angular speed and electromagnetic torque for a DC motor with separate winding excitation, which is done by the identification of the plant model using a recurrent high order neural network (RHONN) in discrete-time. Based on the neural identifier model, discrete-time block control and sliding mode techniques are applied; the stability analysis for the whole system using the Lyapunov approach is also included. In order to validate the theoretical analysis, application results are obtained via simulation and finally a prototype is designed and integrated, where real-time experiments are implemented.

Agradecimientos

Agradezco a mis padres, ***Ma. Irene Hernández Santana*** y ***J. Refugio Castañeda Vargas*** por su comprensión, cariño y apoyo que siempre me han brindado y porque siempre me han ayudado a conseguir todo lo que ahora tengo.

Agradezco a mi esposa ***Luz Elena Díaz Jiménez*** por apoyarme en la realización de este trabajo y juntos cumplir nuestros sueños.

Agradezco a mi pequeña hija ***Andrea Castañeda Díaz*** por ser tan cariñosa conmigo y motivarme para ser una mejor persona.

Agradezco a mis hermanos ***Martha Alicia, Dora Velia, J. Refugio Israel*** y ***Annie Edith*** por su comprensión y apoyo que me han otorgado durante todo este tiempo.

Agradezco a mis asesores de tesis ***Dr. Edgar N. Sánchez Camperos***, ***Dr. Bernardino Castillo-Toledo***, y al ***Dr. Alexander G. Loukianov*** por sus conocimientos compartidos, el apoyo brindado, así como por la paciencia que me tuvieron durante la elaboración de esta tesis.

Agradezco a mis sobrinos ***Ulises, Héctor Hugo, Cinthya, Israel Alejandro, Stephanie*** y ***Alondra Itzel*** por representar una inspiración para realizar este trabajo.

Agradezco al ***CINVESTAV*** por permitirme realizar mis estudios de doctorado y muy especialmente al ***CONACYT*** por el apoyo económico.

Agradezco a la ***Universidad de Guadalajara -Centro Universitario de los Lagos*** (***CULAGOS***) por brindarme el apoyo para la realización del presente trabajo.

Agradezco a mis amigos ***Miguel Hernández González, Onofre Amador Morfín Garduño, Fernando Ornelas Tellez, Ramón García Hernández, Ma. Guadalupe Ciprian Arceo, Adan Landa Hernández, Rubén Machucho Cadena, Miguel Bernal Marín, Alejandro Pizano Moreno, Alejandro Cervantes Herrera*** y ***Adrian Navarro Díaz,*** por su amistad, ayuda y comprensión durante el tiempo en que convivimos juntos.

A ***Jorge Alcantar*** y ***Gaspar González Briceño*** por su colaboración para realizar los experimentos en tiempo real.

Acknowledgements

I would like to thank:

My parents ***Ma. Irene Hernández Santana*** and ***J. Refugio Castañeda Vargas*** for their affection and support, which have always offered to me; they have always helped me to obtain everything which I have.

My wife ***Luz Elena Díaz Jiménez*** for supporting me and together fulfilling our dreams.

My little daughter ***Andrea Castañeda Díaz*** for being so affectionate with me and for motivating me to be one better person.

My brothers and sisters ***Martha Alicia, Dora Velia, J. Refugio Israel*** and ***Annie Edith*** for their comprehension and support during all the time.

My advisors ***Dr. Edgar N. Sánchez Camperos***, ***Dr. Bernardino Castillo-Toledo***, and ***Dr. Alexander G. Loukianov*** by their shared knowledge, the offered support, as well as by their patience during the elaboration of this disseration.

My nephews and nieces ***Ulises, Héctor Hugo, Cinthya, Israel Alejandro, Stephanie*** and ***Alondra Itzel*** to represent an inspiration for finishing this work.

CINVESTAV for accepting me to develop PhD studies and especially ***CONACYT*** for granting the scholarship, which allows me to develop this dissertation.

Universidad de Guadalajara -Centro Universitario de los Lagos (***CULAGOS***) for offering to me the support for the accomplishment of the present work.

My friends ***Miguel Hernández González, Onofre Amador Morfín Garduño, Fernando Ornelas Tellez, Ramón García Hernández, Ma. Guadalupe Ciprian Arceo, Adan Landa Hernández, Rubén Machucho Cadena, Miguel Bernal Marín, Alejandro Pizano Moreno, Alejandro Cervantes Herrera*** and ***Adrian Navarro Díaz*** for their friendship, support and understanding during the time of developing this dissertation.

To ***Jorge Alcantar*** and ***Gaspar González Briceño*** for their collaboration to implement real time experiments.

Contents

List of Figures

Chapter 1

Introduction.

This chapter presents a brief description of the preliminary knowledge required to develop this dissertation. Additionally, the motivation, the objectives and the dissertation outline are also included.

1.1 Preliminaries

DC motors with separate winding excitation play an important role in industry due to its versatility for connection and control. For variable speed and torque applications, the objective is to manipulate the control inputs in such a manner as to make the motor outputs to track a trajectory, requiring a rapid recovery from speed drop caused by a load disturbance. In a DC motor the magnetic field is always at right angle with the field created by the armature winding; this condition is needed to generate maximum torque. The field and armature windings are excited from separate sources; hence, the DC motor with separate winding excitation is a nonlinear multi-input multi-output (MIMO) plant, with two control inputs (armature and field voltages), two controlled outputs (angular speed and field current) or (the electromagnetic torque controlling the armature and field currents) and one disturbance (the load torque). Provided that the state variables are measured and the plant parameters are known, different controllers have been proposed in continuous-time, for example sliding mode controllers ([1], [5], [36], [44], [45]) and passivity-based ones [23]. Furthermore, there exist no many publications for controlling the electromagnetic torque; the only available is [17], to the best of our knowledge. Besides, it is known that the sliding mode controllers ensure robustness of the closed-loop system with respect to match perturbations [8]. In practice, many control plants dynamics have both, matched and unmatched perturbations (for example electrical drives) due to many parameter variations and external disturbances (load torque).

On the other hand, a neural network (NN) is a massively parallel distributed processor which can store experimental knowledge and makes it available for use [14]. It resembles the brain in two respects:

1. Knowledge is acquired by the network through a learning process.
2. Interneuron connection strengths known as synaptic weights are used to store the knowledge.

Artificial NN are systems inspired from how the human brain works. They consist of a finite number of neurons, which are interconnected to each other. The strength of the connections is quantified by means of synaptic weights. The property of primary significance for a NN is its ability to learn from its environment, and to improve its performance through learning [34]. Other basic feature of neural architectures is that they work in parallel. Besides, artificial NN can perform human brain-like tasks such as object and pattern recognition or associative memory. Artificial NN have also provided good solutions to many problems in various fields: such as classification, vision, speech, signal processing, time series prediction, modeling and control, robotics, optimization, experts systems, financial applications, and filtering, among others ([24], [47]). Therefore, artificial NN have become an attractive tool which can be used to construct a model of complex nonlinear processes [24], as exemplified by their applications to identification and control of nonlinear and complex systems [11]; the use of high order neural networks for identification and control has recently been increased [10]. Other interest for studying NN is because they have very good capability in adaptability and learning in presence of external disturbances and uncertainties in modeling.

It is known that for many nonlinear systems, it is often a challenge to obtain their accurate and reliable mathematical models, regarding to their physically complex structures and hidden parameters, as discussed in [7]. Therefore, system identification becomes an important problem and even necessary before system control can be considered not only for understanding and predicting the behavior of the system, but also to obtain an effective control law. Then, the neural identification problem consists in selecting an appropriate neural identification model, and in adjusting its parameters according to an adaptive law, such that the response of the neural identifier model to an input signal (or class of input signals), approximates the response of the real system for the same input [35].

The best well-known training approach for recurrent neural networks (RNN) is the back propagation through time learning [41]. However, it is a first order gradient descent method and hence its learning speed could be very slow [41]. Recently the Extended

Kalman Filter (EKF) based algorithms have been introduced to train neural networks, in order to improve the learning convergence [41]. EKF training of neural networks, both feedforward and recurrent ones, have proven to be reliable and practical for many applications over the past twelve years [41]. There already exist publications about trajectory tracking using neural networks ([10], [35], [37]); in most of them, the design methodology is based on the Lyapunov approach. However, most of those works were developed for continuous-time systems. Moreover, discrete-time neural networks are better fitted for real-time implementations.

There are different alternative approaches that could be followed in the real-time implementations using a digital computer in control systems [16]. One of them is to obtain a continuous-time process model and then to synthesize a continuous-time controller using well-established design methodologies; and then, to discretize the continuous-time controller. Other one is to obtain a discrete-time process model directly by identification methods or via time discretization of a continuous-time model and, then directly synthesize a discrete-time controller; this controller is implemented digitally using rapid sampling, by applying the methodological principles of the so called discrete-equivalent design. For nonlinear discrete-time systems, the real-time implementations control problem is complex due to couplings among subsystems, inputs and outputs [11].

In recent adaptive and robust control literature, numerous approaches have been proposed for the design of nonlinear control systems. Among those, block control (BC) combined with sliding mode (SM) constitutes a well suited design methodology ([26], [27]). Nevertheless, as well as several feedback linearization schemes, the BC technique requires non-singularity of parameters in order to preserve controllability of NN. In this dissertation, we use an EKF-based training algorithm for a recurrent high order neural network (RHONN), in order to identify the model plant and to overcome the controller singularity problem; based on this neural identifier model, a discrete-time SM control law is derived. The block control approach is used to design a nonlinear sliding manifold such that the resulting sliding mode dynamics are described by a desired linear system. The proposed neural identifier and block control applicability are illustrated by angular speed and electromagnetic torque trajectory tracking for a DC motor, via simulations and a prototype is completely integrated, where real-time experiments are implemented.

1.2 Motivation

According to the facts above exposed, it is a necessity to synthesize control algorithms for a class of MIMO discrete-time nonlinear systems (particularly for a DC motor with

separate winding excitation) based on a RHONN. These algorithms must be robust in presence of external disturbances and parameters variations. It is also necessary to test the applicability of the theoretical analysis with real-time experiments.

1.3 Objective

To control the angular speed and the electromagnetic torque of the DC motor with separate winding excitation. In order to achieve this objective, in this dissertation the following steps are considered:

1. To develop a neural identifier in the nonlinear controllable form based on a discrete-time recurrent high order neural network for a DC motor with separate winding excitation.

2. To train the neural identifier with the extended Kalman filter algorithm.

3. To synthesize a discrete-time control law for the DC motor based on the neural identifier with the block control and sliding modes techniques.

4. To design and build a prototype to test in real-time the neural networks control algorithm for a DC motor with separate winding excitation.

1.4 Dissertation Outline

This dissertation presents a solution of trajectory tracking for unknown nonlinear systems (particularly for a DC motor with separate winding excitation). Two cases are considered: the first one, is the angular speed and field current tracking; the second one, is the electromagnetic torque tracking. For both cases, an indirect design method is considered, which is solved with the block control and sliding modes control techniques under complete access to the state vector. The proposed schemes are developed in discrete-time. Moreover, the stability analysis for the whole system is developed. Finally, to test the functionality of the proposed schemes, experimental tests are done in a prototype. All of these results are original and constitute the main contribution of this dissertation.

The dissertation is organized as follows:

Chapter 2 deals with the main theoretical required fundamentals, including mathematical background, the plant model to be controlled and its representation in continuous and discrete-time. We also introduce the neural network structure to identify the plant

model and its respective training algorithm; additionally, it is explained the nonlinear controllable form of the neural network and the applicability of the block control and sliding modes techniques. Finally, the stability analysis for the whole system is developed.

Chapter 3 presents the prototype which is integrated in order to perform the experimental tests, as well as the respective data of the parts included in this prototype.

Chapter 4 first, proposes the RHONN to identify the plant model; then, the EKF algorithm is applied for training the neural network identifier. Additionally, it is explained the application of the block control and discrete-time sliding modes techniques, in order to control the angular speed and the electromagnetic torque of a DC motor with separate winding excitation. Finally, simulations and real-time implementation results are presented.

Chapter 5 presents the conclusions and future work.

Appendix A deals with the conditions required for the block controllable form.

Appendix B lists publications related of this dissertation.

Chapter 2

Fundamentals and Control Scheme

In this chapter, we present the main theoretical fundamentals used in this dissertation, including mathematical background, the plant model to be controlled and its representation in continuous and discrete-time. We also introduce the neural network structure to identify the plant model and its respective training algorithm. Additionally, it is explained the nonlinear controllable form of the neural network and the applicability of the block control and sliding modes techniques. Finally the stability analysis for the whole system is developed.

2.1 Mathematical Preliminaries

In this section, important mathematical preliminaries, required in future chapters, are presented; it close follows [11]. Through this dissertation, we use k as the sampling step, $k \in 0 \cup \mathbb{Z}^+$ $|\bullet|$ as the absolute value and, $\|\bullet\|$ as the Euclidean norm for vectors and as any adequate norm for matrices.

Consider a MIMO nonlinear system:

$$\chi(k+1) = F(\chi(k), u(k), k) \tag{2.1}$$

$$y(k) = h(\chi(k)) \tag{2.2}$$

where $\chi \in \Re^n$, $u \in \Re^m$, and $F \in \Re^n \times \Re^m \rightarrow \Re^n$ is nonlinear function.

Definition 1 *[11] The system* (2.1) *is said to be forced, or to have input. In contrast the system described by an equation without explicit presence of an input u, that is*

$$\chi(k+1) = F(\chi(k))$$

is said to be unforced. It can be obtained after selecting the input u as a feedback function of the state

$$u(k) = \vartheta(\chi(k)) \tag{2.3}$$

Such substitution eliminates u:

$$\chi(k+1) = F(\chi(k), \vartheta(\chi(k))) \tag{2.4}$$

and yields an unforced system (2.4)[18].

Definition 2 *[11] The solution of (2.1)−(2.3) is semiglobally uniformly ultimately bounded (SGUUB), if for any Ω, a compact subset of $\Re^n$ and all $\chi(k_0) \in \Omega$, there exists an $\varepsilon > 0$ and a number $N(\varepsilon, \chi(k_0))$ such that $\|\chi(k)\| < \varepsilon$ for all $k \geq k_0 + N$.*

In other words, the solution of (2.1) is said to be SGUUB if, for any a priory given (arbitrarily large) bounded set Ω and any a priory given (arbitrarily small) set Ω_0, which contains $(0,0)$ as an interior point, there exists a control (2.3) such that every trajectory of the closed loop system starting from Ω enters the set $\Omega_0 = \{\chi(k) \mid \|\chi(k)\| < \varepsilon\}$, in a finite time and remains in it thereafter.

Theorem 1 *[11] Let $V(\chi(k))$ a Lyapunov function for the discrete-time system (2.1), which satisfies the following properties:*

$$\begin{aligned} \gamma_1(\|\chi(k)\|) &\leq V(\chi(k)) \leq \gamma_2(\|\chi(k)\|) \\ V(\chi(k+1)) - V(\chi(k)) &= \Delta V(\chi(k)) \\ &\leq -\gamma_3(\|\chi(k))\| + \delta_0 \end{aligned}$$

where δ_0 is a positive constant, $\gamma_1(\bullet)$ and $\gamma_2(\bullet)$ are strictly increasing functions, and $\gamma_3(\bullet)$ is a continuous, nondecreasing function. Thus if $\Delta V(\chi(k)) < 0$ for $\|\chi(k)\| > \delta_0$, then $\chi(k)$ is uniformly ultimately bounded, i.e. there is a time instant k_T, such that $\|\chi(k)\| < \delta_0$, $\forall k < k_T$.

Definition 3 *[18] A subset $S \in \Re^n$ is bounded if there is $r > 0$ such that $\|\chi\| \leq r$ for all $\chi \in S$.*

Theorem 2 *(Separation Principle) [22]: The asymptotic stabilization problem of the system (2.1)-(2.2), via estimated state feedback*

$$\begin{aligned} u(k) &= \vartheta(\widehat{\chi}(k)) \\ \widehat{\chi}(k+1) &= F(\widehat{\chi}(k), u(k), y(k)) \end{aligned} \tag{2.5}$$

is solvable if and only if, the system (2.1)-(2.2) is asymptotically stabilizable and exponentially detectable.

Corollary 1 *[22] There is an exponential observer for a Lyapunov stable discrete-time nonlinear system (2.1)-(2.2) with $u = 0$ if, and only if, the linear approximation*

$$\begin{aligned} \chi(k+1) &= A\chi(k) + Bu(k) \\ y(k) &= C\chi(k) \end{aligned} \tag{2.6}$$

$$A = \left.\frac{\partial F}{\partial \chi}\right|_{\chi=0} \qquad B = \left.\frac{\partial F}{\partial u}\right|_{\chi=0} \qquad C = \left.\frac{\partial h}{\partial x}\right|_{\chi=0}$$

of the system (2.1)-(2.2) is detectable.

2.2 The Direct-Current Machine

The Direct-current machine is extensively used in industry. It can be controlled over a wide range relatively easy. Large DC machines (in tens or hundreds of horse-power) are used in machine tools, printing presses, conveyors, fans, pumps, hoists, cranes, paper mills, rolling mills, and so forth [39]. In a DC machine, the armature winding is placed on the rotor and the field windings are placed on the stator. When the field and armature windings are supplied from separate voltage sources, the device may operate as either a motor or a generator; it is a motor if it is driving a torque load and a generator if it is being driven by a type of prime mover. In the present work, the DC machine is used as a motor and is connected as separate winding excitation. The Fig. 2.1 shows the equivalent circuit of a DC machine. The equations of voltage and torque are shown below.

2.2.1 Voltage and Torque Equations

The voltage equations for the field winding and rotor coil are [20]

$$u_f = R_f i_f + \frac{d\lambda_f}{dt} \tag{2.7}$$

$$u_a = R_a i_a + \frac{d\lambda_a}{dt} \tag{2.8}$$

where R_a and R_f are the resistance of the armature coil and field winding, respectively; i_a is the armature current; i_f is the field current; λ_f is the field flux linkage; λ_a is the armature flux linkage, u_f and u_a are the field winding and rotor coil voltages, respectively. The expressions for the flux linkage are

$$\lambda_f = L_f i_f + L_{fa} i_a \tag{2.9}$$

$$\lambda_a = L_{af} i_f + L_a i_a \tag{2.10}$$

where L_f and L_a are the inductances of the field winding and armature coil, respectively; $L_{af} = L_{fa}$ is the mutual inductance between the field winding and armature coil.

The mutual inductance between the field winding and armature coil is

$$L_{af} = -L\cos\theta_r \tag{2.11}$$

where L is constant.

Substituting (2.10) and (2.11) in (2.8), we obtain:

$$u_a = R_a i_a + \frac{d}{dt}\left(-L\cos\theta_r i_f + L_a i_a\right) \tag{2.12}$$

and

$$u_a = R_a i_a + \omega_m L i_f \sin\theta_r \frac{d\theta_r}{dt} - L\cos\theta_r + L_a \frac{di_a}{dt} \tag{2.13}$$

For a machine with commutator, the pseudo stationary coil L_{af} is constant for the position of the brushes $\theta_r = \pi/2$ and $\omega_m = \frac{d\theta_r}{dt}$; then

$$u_a = R_a i_a + L_{af}\omega_m i_f + L_a \frac{di_a}{dt} \tag{2.14}$$

Then, according to the equivalent circuit of a DC-machine represented in Fig. 2.1, the field and armature voltage equations can be represented in the following form [20]:

$$\begin{bmatrix} u_f \\ u_a \end{bmatrix} = \begin{bmatrix} R_f + \rho L_f & 0 \\ L_{af}\omega_m & R_a + \rho L_a \end{bmatrix} \begin{bmatrix} i_f \\ i_a \end{bmatrix} \tag{2.15}$$

with $\rho = d/dt$, ω_m is the angular speed.

The voltage induced in the armature circuit is $u_a = L_{af}\omega_m i_f$, and is commonly referred to as the counter or back emf. It also represents the open-circuit armature voltage.

The electromagnetic torque [20] is given by

$$T_e = L_{af} i_a i_f \tag{2.16}$$

Note that the electromagnetic torque depends on the value of L_{af}. Clearly if L_{af} is uncertain, this induces an uncertainty also in the value of T_e.

The torque and angular speed are related by

$$T_e = J_e \frac{d\omega_m}{dt} + b_e\omega_m + T_L \tag{2.17}$$

where J_e is the inertia of the rotor ($kg \cdot m^2$ or $J \cdot s^2$); b_e is a damping coefficient associated with the mechanical rotational system of the machine ($N \cdot m \ \ s$); T_L is the load torque ($N \cdot m$); and $\frac{d\omega_m}{dt}$ is the acceleration of the machine ($m \cdot s^{-2}$).

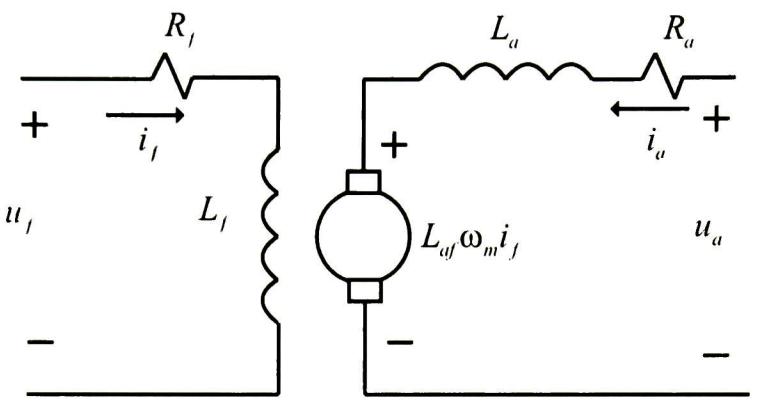

Figure 2.1: Equivalent circuit of a DC-machine

2.2.2 DC Motor Model

The state space equations of the DC motor can be obtained with the direct manipulation of the field and armature voltages equations (2.15) and the torque and angular speed relation (2.17). Then the third order continuous-time DC motor model is given by [20]

$$\begin{aligned} \dot{\omega}_m &= -\frac{b_e}{J_e}\omega_m + \frac{L_{af}}{J_e} i_a i_f - \frac{1}{J_e} T_L \\ \frac{di_a}{dt} &= -\frac{R_a}{L_a} i_a - \frac{L_{af}}{L_a}\omega_m i_f + \frac{1}{L_a} u_a \\ \frac{di_f}{dt} &= -\frac{R_f}{L_f} i_f + \frac{1}{L_f} u_f \end{aligned} \tag{2.18}$$

where ω_m is the angular speed; i_a is the armature current, i_f is the field current, u_a is the applied armature voltage; u_f is the applied field voltage; b_e is the equivalent viscous friction coefficient of motor and load referred to motor output shaft; J_e is the equivalent moment of inertia of motor armature and load referred to motor output shaft; L_{af} is the mutual inductance between the field winding and the armature coil; T_L is the load torque; R_a is the armature resistance; L_a is the armature inductance; R_f is the field resistance and L_f is the field inductance.

The DC motor with separate wind excitation has two control inputs (the armature and field voltage), two controlled outputs (the angular speed and the field current or the electromagnetic torque) and one disturbance (the load torque). For electromagnetic torque or angular speed control applications, the objective is to manipulate the inputs in such a manner as to force the motor outputs to track a desired trajectory.

2.2.3 Discrete-time DC Motor Model

For implementing a digital computer-control system, a time discretization procedure has to be applied to the continuous-time system. We first put in evidence the discrete-time structure of the discretized model. This model (2.18) can be described using the Euler method [16], as

$$\begin{aligned} \omega_m(k+1) &= \omega_m(k) + T_s\left(-\frac{b_e}{J_e}\omega_m(k) + \frac{L_{af}}{J_e}i_a(k)i_f(k) - \frac{1}{J_e}T_L(k)\right) \\ i_a(k+1) &= i_a(k) + T_s\left(-\frac{R_a}{L_a}i_a(k) - \frac{L_{af}}{L_a}\omega_m(k)i_f(k) + \frac{1}{L_a}u_a(k)\right) \\ i_f(k+1) &= i_f(k) + T_s\left(-\frac{R_f}{L_f}i_f(k) + \frac{1}{L_f}u_f(k)\right) \end{aligned} \tag{2.19}$$

where $k \in Z^+ \cup 0$ is the discrete time with Z^+ the set of nonnegative numbers, and T_s is the sampling period.

We will propose a new structure on the basis of the first approximation. The reason behind this choice is that one of the interesting properties of NN is that the adjusting of their parameters to compesate the effects of the neglected terms.

2.3 Neural Networks

Recently, NN have become an attractive tool which can be used to construct a model of complex nonlinear processes. This is due to NN have an inherent ability to learn and approximate a nonlinear function arbitrarily well. A large number of identification and control structures have been proposed on the basis of neural networks in recent years [24].

In general, it is possible to define four different classes of neural architectures [14], as: single and multilayer feedforward networks, lattice structures and recurrent ones. The last ones are described below.

2.3.1 Recurrent High Order Neural Networks

They distinguish from a feedforward neural network, for having have at least one feedback loop. For example, a recurrent network may consist of a single layer of neurons with each neuron feeding its output signal back to the inputs of all the other neurons, as illustrated in Fig. 2.2, where there are no self-feedback loops in the network. The recurrent network illustrated in Fig. 2.2 also has no hidden neurons. In Fig. 2.3 we illustrate another class of recurrent networks with hidden neurons. The feedback connections shown

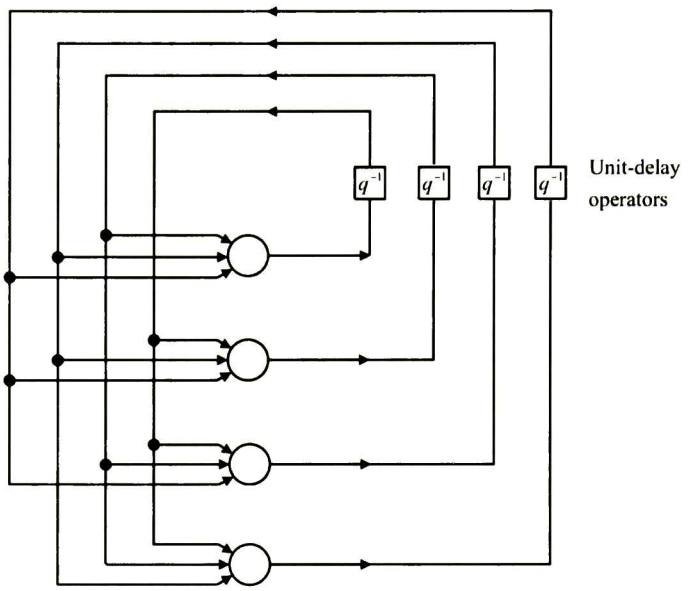

Figure 2.2: Recurrent network with no self-feedback loops and no hidden neurons

in Fig. 2.3 originate from the hidden neurons as well as from the output neurons. The presence of feedback loops, as in Fig. 2.2 or as in Fig. 2.3, has a profound impact on the learning capability of the network, and on its performance [14]. Moreover, the feedback loops involve the use of particular branches composed of unit-delay elements (denoted by q^{-1}), which result in a nonlinear dynamical behavior by virtue of the nonlinear nature of the neurons. The neural networks described previously, are denominated of first order because the entrance to a neuron represents the linear combination of its components. In a recurrent second order neural network, the total input to the neuron is not only a linear combination of the components, but also of their products. Moreover, one can pursue along this line and include high-order interactions of high order. This kind of neural networks are called recurrent high order neural networks (RHONN) [19]. The RHONN model is flexible and allows to incorporate to the neural model a priory information about the system structure with less units.

2.3.2 Neural Identification

It is known that for many nonlinear systems it is often difficult to obtain their accurate and reliable mathematical model, due to their physically complex structures and hidden

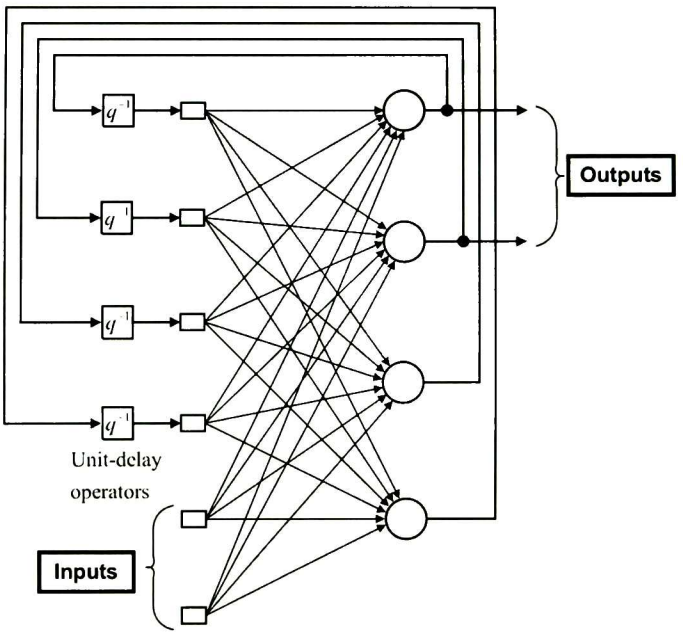

Figure 2.3: Recurrent network with hidden neurons

parameters, as is discussed in [7]. Therefore, system identification becomes an important problem and even necessary before system control can be considered not only for understanding and predicting the behavior of the system, but also to obtain an effective control law. The neural identification problem consist of the selection of an appropriate neural identification model and adjusting its parameters according to an adaptive law, such the response of the model to an input signal (or class of inputs signals), approximates the response of the real system for the same input [35].

The neural identification scheme, which is used in the present work, is illustrated in Fig. 2.4.

2.3.3 Discrete-time Recurrent High Order Neural Networks

Consider the following discrete-time RHONN:

$$x_i(k+1) = w_i^T z_i\left(\chi(k), u(k)\right), \; i = 1, \cdots, n \tag{2.20}$$

where x_i, $i = 1, \cdots, n$ is the state of the ith neuron, $\chi(k)$ is the plant state, L_i is the respective number of high-order connections, $\{I_1, I_2, \cdots, I_{L_i}\}$ is a collection of non-ordered

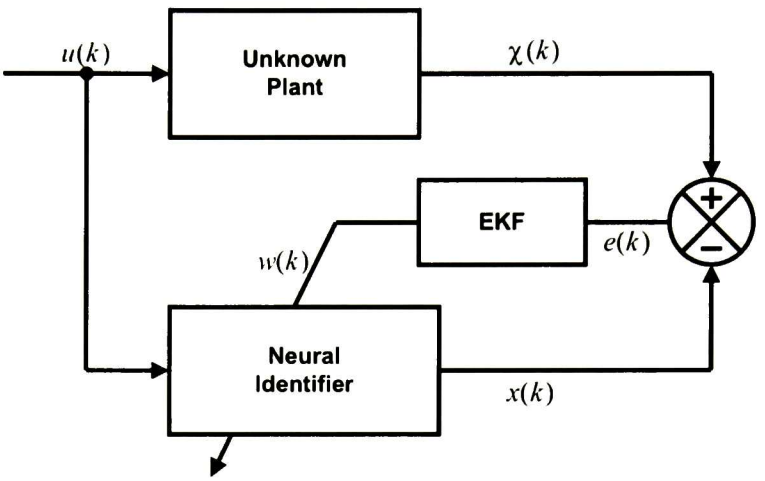

Figure 2.4: Neural identification scheme

subsets of $\{1, 2, \cdots, n\}$, n is the state dimension, w_i, $i = 1, \cdots, n$ is the respective on-line adapted weight vector, and $z_i(\chi(k), u(k))$ is given by

$$z_i(\chi(k), u(k)) = \begin{bmatrix} z_{i_1} \\ z_{i_2} \\ \vdots \\ z_{i_{L_i}} \end{bmatrix} = \begin{bmatrix} \prod_{j \in I_1} y_{i_j}^{d_{ij}(1)} \\ \prod_{j \in I_2} y_{i_j}^{d_{ij}(2)} \\ \vdots \\ \prod_{j \in I_i} y_{i_j}^{d_{ij}(i)} \end{bmatrix} \tag{2.21}$$

with d_{i_j} being a nonnegative integers, and y_i is defined as follows:

$$y_i = \begin{bmatrix} y_{i_1} \\ \vdots \\ y_{i_1} \\ y_{i_{n+1}} \\ \vdots \\ y_{i_{n+m}} \end{bmatrix} = \begin{bmatrix} S(\chi_1(k)) \\ \vdots \\ S(\chi_n(k)) \\ u_1(k) \\ \vdots \\ u_m(k) \end{bmatrix} \tag{2.22}$$

In (2.22), $u(k) = [u_1(k), ..., u_m(k)]^T$ is the input vector to the neural network, and $S(\bullet)$ is defined by

$$S(\chi(k)) = \frac{1}{1 + exp(-\beta\chi(k))} \tag{2.23}$$

Note that in (2.22), $u(k)$ is not affected by sigmoidal function (2.23), this is due to the fact to have an affine control.

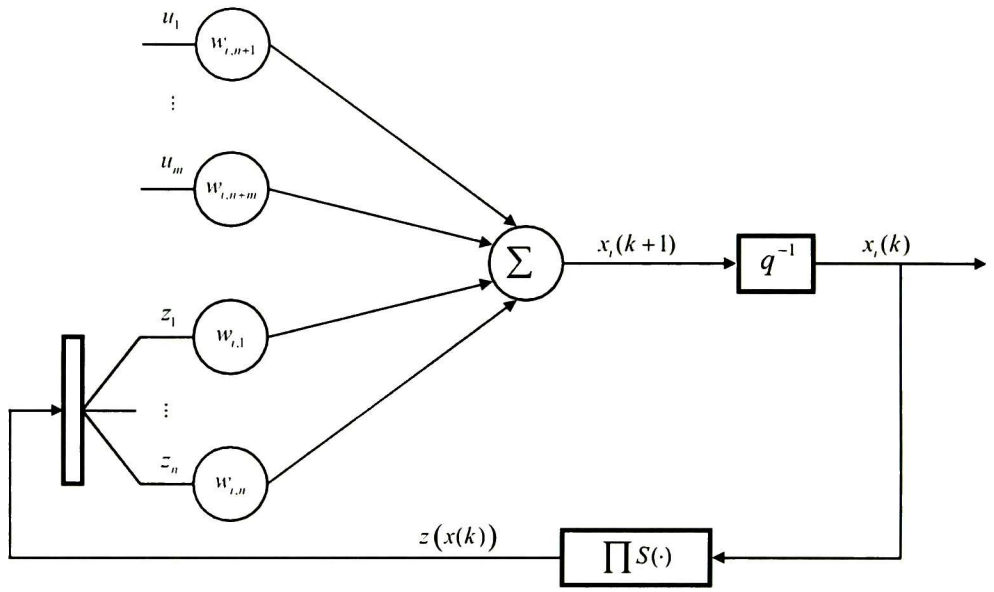

Figure 2.5: Scheme of a discrete-time RHONN

Fig. 2.5 shows the scheme of a discrete-time RHONN.
From (2.20) three possible models of RHONN can be derived:

- *Parallel model.* In this configuration, the feedback connections of the NN are from the NN outputs.

$$x_i(k+1) = w_i^T z_i(x(k), u(k)), \quad i = 1, \cdots, n \tag{2.24}$$

- *Series-Parallel model.* In this configuration, the feedback connections of the NN are from the real plant outputs.

$$x_i(k+1) = w_i^T z_i(\chi(k), u(k)), \quad i = 1, \cdots, n \tag{2.25}$$

- *Feedforward model (HONN).* In this configuration, the connections of the NN are from the input signals.

$$x_i(k) = w_i^T z_i(u(k)), \quad i = 1, \cdots, n \tag{2.26}$$

where $x(k)$ is the NN state vector, $\chi(k)$ is the plant state vector and $u(k)$ is the input vector to the NN.

In the present dissertation, we use the series-parallel configuration (2.25) due to the fact that this model constitutes a well approximation method of the real plant by the neural identifier.

Consider the problem to approximate the general discrete-time nonlinear system (2.1), by the following discrete-time RHONN series-parallel representation [35]

$$x_i(k+1) = w_i^{*T} z_i\left(\chi(k), u(k)\right) + \epsilon_{z_i}, \quad i = 1, \cdots, n \tag{2.27}$$

where χ_i is the ith plant state, ϵ_{z_i} is a bounded approximation error, which can be reduced by increasing the number of the adjustable weights [35]. Assume that there exists ideal weights vector w_i^* such that $\|\epsilon_{z_i}\|$ can be minimized on a compact set $\Omega_{z_i} \subset \Re^{L_i}$ The ideal weight vector w_i^* is an artificial quantity required for analytical purpose [35]. In general, it is assumed that this vector exists and is constant but unknown. Let us define its estimate as w_i and the estimation error as

$$\widetilde{w}_i(k) = w_i(k) - w_i^*. \tag{2.28}$$

The estimate w_i is adjusted by the EKF based algorithm.

2.3.4 The Extended Kalman Filtering Training Algorithm

The Kalman filtering (KF), formulated in the state-space representation of linear dynamical systems, provides a recursive solution to the linear optimal filtering problem. It applies to stationary as well as nonstationary enviroments. The solution is recursive because each updated estimate of the state is computed from the previous one and the new input data; hence, only the previous estimate requires storage. In addition to eliminating the need to storing the entire past observe data, the KF is computationally more efficient than computing the estimate directly from the entire past observed data, at each step of the filtering process [13]. The KF estimates the state of a linear system with additive state and output noises [12], [42]. The KF has been proposed for the training of the NN. The main features for KF-based neural network training are:

1. The network weights become the states to be estimated. In this case, the error between the neural network output and the measured plant output can be considered as additive white noise.

2. The measurement used for the KF is the output of the neural identifier.

3. The training goal is to find the optimal weight values which minimize the prediction error.

NN training results on a nonlinear systems; hence, the Extended Kalman Filter (EKF) is required.

Consider a special case of the MIMO system (2.1) in discrete-time, represented in state space as:

$$\chi(k+1) = F(\chi(k), u(k), k) + q(k) \tag{2.29}$$

$$y(k) = h(\chi(k)) + v(k) \tag{2.30}$$

where $\chi(k)$ is the state of the plant; $u(k)$ is the input to the system; $q(k)$ and $v(k)$ are independent noises of process and measurement respectively; $F(\bullet)$ is the nonlinear transition matricial function, possibly time-varying; $h(\bullet)$ is the nonlinear measurement function, possibly time-varying too; the idea is to linearize equations (2.29)-(2.30) at each instant of time around the most recently estimation. Once the system is linearized, the equations of KF are applied. Therefore, the training of NN is visulized as an optimal filtering problem, where a recursive solution is required, and is not necessary to store all the evolution weights; this is the essence to use the KF for neural networks training.

In this dissertation, we use the EKF-based training algorithm described by

$$w_i(k+1) = w_i(k) + \eta_i K_i(k) e_i(k) \tag{2.31}$$

$$K_i(k) = P_i(k) H_i(k) M_i(k) \tag{2.32}$$

$$P_i(k+1) = P_i(k) - K_i(k) H_i^T P_i(k) + Q_i(k)$$

where $i = 1, \cdots, n$ and with

$$M_i(k) = \left[R_i(k) + H_i^T(k) P_i(k) H_i(k)\right]^{-1} \tag{2.33}$$

$$e_i(k) = \chi_i(k) - x_i(k) \tag{2.34}$$

$e_i(k) \in \Re$ is the respective identification error, $P_i(k) \in \Re^{L_i \times L_i}$ is the prediction error covariance matrix at step k, $w_i(k) \in \Re^{L_i}$ is the weight (state) vector, L_i is the respective number of neural network weights, $\chi_i(k)$ is the ith plant state, $x_i(k)$ is the ith neural network state, n is the number of states, $K_i(k) \in \Re^{L_i}$ is the Kalman gain vector, $Q_i(k) \in \Re^{L_i \times L_i}$ is the NN weight estimation noise covariance matrix, $R_i(k) \in \Re$ is the error noise covariance; $H_i \in \Re^{L_i}$ is a vector, in which each entry (H_{ij}) is the derivative of one of the neural network state $(x_i(k))$, with respect to one neural network weight, $(w_{ij}(k))$, as follows

$$H_{ij}(k) = \left[\frac{\partial x_i(k)}{\partial w_{ij}(k)}\right]^T_{w_i(k)=w_i(k+1)} \tag{2.35}$$

where $i = 1, \cdots, n$ and $j = 1, \cdots L_i$. Usually P_i and Q_i are initialized as diagonal matrices, with entries $P_i(0)$ and $Q_i(0)$, respectively. It is important to remark that $H_i(k)$, $K_i(k)$ and $P_i(k)$ for the EFK are bounded; for a detailed explanation of this fact see [42].

2.4 Nonlinear Block Controllable Form with Disturbances

In this section, a discontinuous control strategy will be develop for a class of nonlinear systems in the Nonlinear Block Controllable (NBC) form.

Consider a discrete-time nonlinear system

$$x(k+1) = f(x(k)) + B(x(k))u(k) + d(\omega(k)) \tag{2.36}$$

$$y(k) = h(x(k)) \tag{2.37}$$

where $k \in Z$, is a subset of the real numbers called the sampling instants, the state vector $x(k)$ is defined on a neighborhood X of the origin of $\Re^n$, $u(k) \in \Re^m$ is the input vector, $y(k) \in \Re^p$ is the output vector. The vector $f(x(k))$, the columns of $g(x(k))$ and $d(\omega(k))$ are smooth vector fields of class $C^{\infty}_{[t,\infty)}$, and in addition, it is assumed that $f(0) = 0$, $h(0) = 0$, and that $d(\omega(k))$ is a known perturbation term that could result from modelling errors, aging and disturbances, which exist in any realistic problem. This assumption will be removed later on. On the other hand, the output tracking error is defined as the difference between the output of the system, y_k, and a reference signal to be tracked, $g(\omega(k))$, i.e.,

$$e(k) = y(k) - g(\omega(k)) \tag{2.38}$$

The reference signal, $g(\omega(k))$, is generated by an external system described by

$$\omega(k+1) = s(\omega(k)), \quad \omega(k) \in \Re^s \tag{2.39}$$

By means of non-singular transformations, system (2.36) is represented in NBC form

as follows:

$$\begin{aligned}
x_1(k+1) &= f_1(x_1(k)) + B_1(x_1(k))x_2(k) + d_1(\omega(k)) \\
x_2(k+1) &= f_2(x_1(k), x_2(k)) + B_2(x_1(k), x_2(k))x_3(k) + d_2(\omega(k)) \\
&\vdots \\
x_i(k+1) &= f_i(x_1(k), \ldots, x_i(k)) + B_i(x_1(k), \ldots, x_i(k))x_{i+1}(k) + d_i(\omega(k)) \qquad (2.40) \\
&\vdots \\
x_r(k+1) &= f_r(x_1(k), \ldots, x_r(k)) + B_r(x_1(k), \ldots, x_r(k))u_k + d_r(\omega(k)) \\
(i &= 3, \ldots, n-1) \\
y(k) &= x_1(k) \qquad (2.41) \\
& x(k) \in \Re^n, \quad u(k) \in \Re^m, \quad y(k) \in \Re^p, \quad \omega(k) \in \Re^s \qquad n_1 = p \leq m < n
\end{aligned}$$

where $x(k) = \left[\begin{array}{ccc} x_1^T(k) & \cdots & x_r^T(k) \end{array} \right]^T$ with $x_j(k) \in \Re^{n_j}$, $j = 1, \ldots, r$; and the set of numbers $(n_1, \ldots, n_r)$ defines the structure of system (2.40) as follows

$$n_1 \leq n_2 \leq \ldots \leq n_r \leq m \qquad (2.42)$$

with the following assumptions:

Assumption 1 *The B_j matrix has full row rank*

$$rank(B_j) = n_j \quad \forall x \in R^n \text{ with } \sum_{j=1}^{r} n_j = n.$$

Assumption 2 *The B_j matrix is bounded*

$$|B_j(x_1(k), \ldots, x_r(k))| \leq \varsigma_j, \quad \varsigma_j > 0.$$

The procedure of reducing the system (2.36) to the NBC form (2.40) based on integral transformation method [28], is presented in Appendix A.

2.5 Control Scheme

The scheme used in the present work is the Neural and Sliding Modes Control, where the control law is developed from neural identification model of the plant. Fig. 2.6 displays the proposed control scheme. This kind of scheme is considered an "indirect control system". The idea is to use a neural network (neural identifier) to built a block control model of the plant whose parameters (weights) are updated on-line. Using these

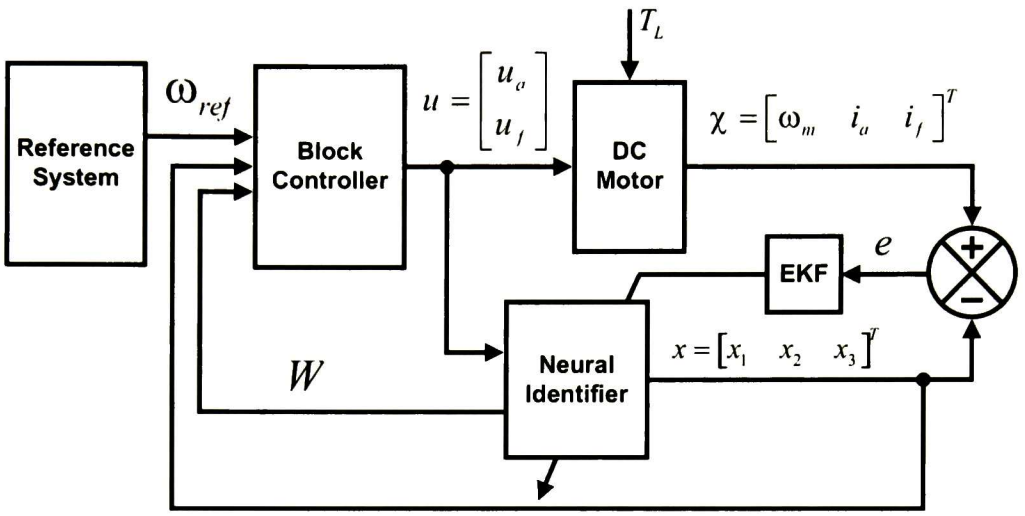

Figure 2.6: Control Scheme

parameters, by means of the sliding modes technique a control law is synthesized. The control objective is to develop the output plant tracks a desired reference signal given by the reference system.

We use a neural model for the next reasons: the parameters and the external disturbance of the plant to be controlled are unknown, then, it is necessary to approximate the response of the plant model with the response of a neural identifier for the same input; we can propose a neural identifier structure based on the plant model structure (in case of the plant structure is known) and use a priory information about the plant for the neural identifier training; based on the neural identifier structure, a particular control technique is applied.

2.5.1 Neural Block Identification

In this sub-section we consider the problem to identify the nonlinear system (2.1) using a RHONN (2.20) trained with the EKF algorithm (2.31)-(2.33) based on the following theorem.

Theorem 3 *The RHONN (2.20) trained with the EKF-based algorithm (2.31)-(2.33) to identify the nonlinear plant (2.1), ensures that the weight estimation error (2.28) and the identification error (2.34) are semiglobally uniformly ultimately bounded (SGUUB); moreover, the RHONN weights remain bounded.*

The respective proof is presented in [2].

In this work, we consider system (2.1)-(2.2) represented in the following triangular form:

$$\begin{aligned} \chi^1(k+1) &= f_1(\bar{\chi}^2(k), k) \\ \chi^i(k+1) &= f_i((\bar{\chi}^{i+1}(k)), k), \ i = 2, ..., r-1 \\ \chi^r(k+1) &= f_r((\chi(k), u(k), k) \\ y(k) &= h(\chi(k)) = \chi^1(k). \end{aligned} \tag{2.43}$$

where $\chi \in \Re^n$ is the state vector of the plant, $\chi = \left[\chi^{1T}\ \chi^{2T} ... \chi^{rT}\right]^T$, $\chi_i \in \Re^{n_i}$; $\bar{\chi}^i = \left[\chi^{1T}\ \ \chi^{2T} ... \chi^{iT}\right]^T$ $y \in \Re^{n_1}$ is the output vector to be controlled. The numbers n_i, $i = 1, ..., r$ define the structure of the plant and satisfy $n_i \leq n_{i+1}$ and $\sum_1^r n_i = n$. It is assumed that

$$rank\left(\frac{\partial f_i(k)}{\partial \chi_{i+1}(k)}\right) = n_i, \forall \chi \in X.$$

It is clear that this assumption is needed for system (2.43) be controllable. Moreover, the control signal requires to be bounded by

$$\|u(k)\| \leq u_0, \quad u_0 > 0. \tag{2.44}$$

Note that the motor model has the structure (2.43) with $\chi^1 = \omega_m$, $\chi^2 = (i_a\ i_f)^T$ $n_1 = 1$, $n_2 = 2$ and $r = 2$.

Given that system (2.43) has the triangular structure, the sliding mode block feedback linearization technique ([10], [26]) can be applied to design a control law. For this, we propose the neural identifier (2.20) in the following NBC form, which corresponds to system (2.43) structure:

$$\begin{aligned} x^1(k+1) &= W_1(k) z_1\left(\chi^1(k)\right) + \bar{W}_1 \chi^2(k) \\ x^i(k+1) &= W_i(k) z_i\left(\bar{\chi}^i(k)\right) + \bar{W}_i \chi^{i+1}(k), \\ i &= 2, ..., r-1 \\ x^r(k+1) &= W_r(k) z_r\left(\chi(k)\right) + \bar{W}_r u(k) \\ \tilde{y}(k) &= \tilde{h}\left(\chi(k)\right) = x_1(k) \end{aligned} \tag{2.45}$$

where $x \in \Re^n$ is the state vector of the neuronal identifier, $x = \left[x^{1T}\ \ x^{2T} ... x^{rT}\right]^T$, $x_i \in \Re^{n_i}$; $\bar{x}^i = \left[x^{1T}\ \ x^{2T} ... x^{iT}\right]^T$ $\tilde{y} \in \Re^{n_1}$ is the output vector; W_i, $i = 1, ..., r$ is the respective on-line adapted weight matrix; $\bar{W}_i$ is a matrix which is required to have constant entries in

order to define the controllability weights and guarantee the controllability of the system [10]; $z_i(\bar{\chi}^i(k))$ is defined as in (2.21).

2.5.2 Control Synthesis

The control synthesis is based on the following proposition:

Proposition 1 *Given a desired output trajectory χ_d^1, the dynamic system (2.43) with output χ^1 and a neural network (2.45) with output x^1, then it is possible to establish the following inequality [10]:*

$$\|\chi_d^1 - \chi^1\| \leq \|x^1 - \chi^1\| + \|\chi_d^1 - x^1\| \tag{2.46}$$

where $(\chi_d^1 - \chi^1)$ is the system output tracking error, $(x^1 - \chi^1)$ is the output identification error and $(\chi_d^1 - x^1)$ is the RHONN output tracking error.

It is possible to establish Proposition 1 due to the Separation Principle for discrete-time nonlinear systems [22], as stated in Theorem 2 and Corollary 1 if in (2.6) $C = I$.

Based on Propostion 1, it is possible to divide the tracking error in two parts [10]:

1. Minimization of $(x^1 - \chi^1)$, which can be achieved by the proposed on-line identification algorithm (2.31)-(2.33) on the basis of Theorem 3.

2. Minimization of $(\chi_d^1 - x^1)$, for that a tracking algorithm is developed on the basis of the neural identifier (2.45).

Following Theorem 3, there exists a bounded vector valued function $\Delta_i(k)$ such that

$$\chi^i(k) = x^i(k) + \Delta_i(k), \ i = 1, ..., r \tag{2.47}$$

with

$$\|\Delta_i(k)\| \leq \zeta_{i1}. \tag{2.48}$$

The control law is now developed based on the neural network identifier (2.45) updated with the EKF algorithm (2.31)-(2.33). Applying the block control technique, we introduce the following iterative transformation:

$$\begin{aligned} \varepsilon^1(k) &= \chi^1(k) - \chi_d^1(k) \\ &= x^1(k) - \chi_d^1(k) + \Delta_1(k), \\ \varepsilon^i(k) &= \chi^i(k) - \chi_d^i(k) \\ &= x^i(k) - \chi_d^i(k) + \Delta_i(k), \quad i = 2, ..., r \end{aligned} \tag{2.49}$$

where χ_d^i is the desired value for χ^i and

$$\begin{aligned} \chi_d^2(k) &= \bar{W}_1^{-1}[\mathbf{K}_1\varepsilon^1(k) - \tilde{f}_1\left(\chi^1(k),k\right)] \\ \chi_d^{i+1}(k) &= \bar{W}_i^{-1}[\mathbf{K}_i\varepsilon^i(k) - \tilde{f}_i\left(\bar{\chi}^i(k),k\right)] \\ i &= 2,...,r, \end{aligned} \tag{2.50}$$

where $\tilde{f}_1\left(\chi^1(k),k\right) = W_1 z_1\left(\chi^1(k)\right) - \chi_d^1(k)$, $\tilde{f}_i\left(\bar{\chi}^i(k),k\right) = W_i z_i\left(\bar{\chi}^i(k)\right) - \chi_d^i(k)$, $i = 2,...,r$ with $\mathbf{K}_i$ a Schur matrix.

Proposition 2 *Transformation (2.49) and (2.50) reduces system (2.45) to the following form:*

$$\begin{aligned} \varepsilon^1(k+1) &= \mathbf{K}_1\varepsilon^1(k) + \bar{W}_1\varepsilon^2(k) + \tilde{\Delta}_1(k) \\ \varepsilon^i(k+1) &= \mathbf{K}_i\varepsilon^i(k) + \bar{W}_i\varepsilon^{i+1}(k) + \tilde{\Delta}_i(k), \\ \varepsilon^r(k+1) &= W_r(k)z_r\left(\chi(k)\right) + \bar{W}_r u(k) \\ &\quad -\chi_d^r(k+1) + \tilde{\Delta}_r(k) \\ i &= 2,\cdots,r-1 \end{aligned} \tag{2.51}$$

where $\tilde{\Delta}_i(k) = \Delta_i(k+1)$, $i = 1,\cdots,r$.

Proof. The procedure close follows the one presented in Appendix A.

Step 1. At this step, deriving the error dynamics (2.49)

$$\varepsilon^1(k) = x^1(k) - \chi_d^1(k) + \Delta_1(k)$$

by using system (2.45) and imposing the desired dynamics $(\mathbf{K}_1\varepsilon^1(k) + \tilde{\Delta}_1(k))$, yields

$$\begin{aligned} \varepsilon^1(k+1) &= x^1(k+1) - \chi_d^1(k+1) + \tilde{\Delta}_1(k) \\ &= \tilde{f}_1(\chi^1(k),k) + \bar{W}_1\chi^2(k) + \tilde{\Delta}_1(k) \\ &= \mathbf{K}_1\varepsilon_1(k) + \tilde{\Delta}_1(k) \end{aligned} \tag{2.52}$$

where $\tilde{f}_1(\chi^1(k),k) = W_1(k)z_1\left(\chi^1(k)\right) - \chi_d^1(k+1)$ and $\mathbf{K}_1$ is a Shur matrix. Then, we calculate the desired value of the virtual control $\chi^2(k)$ from (2.52) of the form

$$\chi_d^2(k) = -\bar{W}_1^{-1}[\tilde{f}_1(\chi^1(k),k) - \mathbf{K}_1\varepsilon_1(k)] \tag{2.53}$$

The second error $\varepsilon^2(k)$ is defined as (2.49)

$$\varepsilon^2(k) = \chi^2(k) - \chi_d^2(k) \tag{2.54}$$

Then substituting

$$\chi^2(k) = \varepsilon_2(k) + \chi_d^2(k)$$

into (2.52), gives the first transformed block (2.51), namely

$$\varepsilon^1(k+1) = \mathbf{K}_1\varepsilon_1(k) + \bar{W}_1\varepsilon_2(k) + \tilde{\Delta}_1(k). \tag{2.55}$$

Step 2. At this step, proceeding in the same way, error $\varepsilon^2(k)$ is represented as

$$\varepsilon^2(k) = x^2(k) - \chi_d^2(k) + \Delta_2(k)$$

having the dynamics

$$\varepsilon^2(k+1) = \tilde{f}_2(\bar{\chi}^2(k), k) + \bar{W}_2\chi^3(k) + \tilde{\Delta}_2(k) \tag{2.56}$$

where $\bar{\chi}^2 = [\chi^{1T} \;\; \chi^{2T}]^T$. $\tilde{\Delta}_2(k) = \Delta_2(k+1)$ and

$$\tilde{f}_2(\bar{\chi}^2(k), k) = W_2(k)z_2\left(\bar{\chi}^2(k)\right) - \chi_d^2(k+1)$$

Now, imposing the desired dynamics of the form

$$\varepsilon^2(k+1) = \mathbf{K}_2\varepsilon^2(k) + \tilde{\Delta}_2(k) \tag{2.57}$$

we calculate the desired value $\chi_d^3(k)$ from (2.56) and (2.57) as

$$\chi_d^3(k) = -\bar{W}_2^{-1}[\tilde{f}_2(\bar{\chi}^2(k), k) - \mathbf{K}_2\varepsilon^2(k)] \tag{2.58}$$

which is the reference value for $\chi^3(k)$. Forming the third control error

$$\varepsilon^3(k) = \chi^3(k) - \chi_d^3(k)$$

we have

$$\chi^3(k) = \varepsilon_3(k) + \chi_d^3(k). \tag{2.59}$$

Using (2.59) and (2.58), the equation (2.56) is rewritten in the new variables $\varepsilon^2(k)$ and $\varepsilon^3(k)$ as

$$\varepsilon^2(k+1) = \mathbf{K}_2\varepsilon^2(k) + \bar{W}_2\varepsilon^3(k) + \tilde{\Delta}_2(k).$$

Step r. Iterating these steps and having finally the desired value $\chi_d^r(k)$, we introduce the last error vector $\varepsilon^r(k)$:

$$\begin{aligned} \varepsilon^r(k) &= \chi^r(k) - \chi_d^r(k) \\ &= x^r(k) - \chi_d^r(k) + \Delta_r(k). \end{aligned}$$

Then the last block of (2.45) can be represented of the form

$$\begin{aligned} \varepsilon^r(k+1) &= W_r(k) z_r\left(\chi(k)\right) + \bar{W}_r u(k) \\ &\quad -\chi_d^r(k+1) + \tilde{\Delta}_r(k). \end{aligned}$$

Thus the (2.49) and (2.50) reduces the system (2.45) to (2.51). ■

The sliding function $s_D\left(k\right)$ can be derived from the block control transformation (2.49), and a natural selection for this function is $s_D\left(k\right) = \varepsilon^r\left(k\right)$. Thus, the last block of system (2.51) is represented as

$$s_D(k+1) = \tilde{f}_r\left(\chi(k), k\right) + \bar{W}_r u(k) + \tilde{\Delta}_r(k) \tag{2.60}$$

with

$$\tilde{f}_r\left(\chi(k), k\right) = W_r(k) z_r\left(\chi(k)\right) - \chi_d^r(k+1). \tag{2.61}$$

To enforce sliding mode on the manifold $s_D = 0$, $u(k)$ can be selected as $u(k) = u_{eq}(k)$ [44] and

$$u_{eq}(k) = -\left[\bar{W}_r\right]^{-1}\left[\tilde{f}_r\left(\chi(k), k\right) + \tilde{\Delta}_r(k)\right] \tag{2.62}$$

where the equivalent control $u_{eq}(k)$ is calculated as the solution of $s_D(k+1) = 0$. This control brings the system trajectory on the sliding manifold $s_D(k) = 0$ in one sampling time period. However, $u_{eq}(k)$ (2.62) can not be implemented since the identification error $\tilde{\Delta}_r(k)$ is unknown. Indeed, it is possible to cancel the only nonlinear known term $\tilde{f}_r\left(\chi(k), k\right)$ in (2.60) by the control

$$u(k) = u_s(k), \qquad u_s(k) = -\left[\bar{W}_r\right]^{-1} \tilde{f}_r\left(\chi(k), k\right) \tag{2.63}$$

remaining the uncertain term $\tilde{\Delta}_r(k)$. Therefore, in order to reduce the effect of $\tilde{\Delta}_r(k)$, we impose the desired dynamics for the sliding variable s_D as ([46], [48])

$$s_D(k+1) = \tilde{\Delta}_r(k) - \tilde{\Delta}_r(k-1). \tag{2.64}$$

The term $\tilde{\Delta}_r(k-1)$ can be calculated from equation (2.60) as

$$\tilde{\Delta}_r(k-1) = s_D(k) - \tilde{f}_r\left(\chi(k-1), k-1\right) - \bar{W}_r u(k-1) \tag{2.65}$$

By inserting (2.65) in (2.64), and then comparing it with equation (2.60), the approximate equivalent control becomes

$$\begin{aligned} u(k) &= \tilde{u}_{eq}(k) \\ \tilde{u}_{eq}(k) &= -\left[\bar{W}_r\right]^{-1}\left[s_D(k) + \tilde{f}_r\left(\chi(k), k\right)\right] \\ &\quad + \left[\bar{W}_r\right]^{-1}\left[\tilde{f}_r\left(\chi(k-1), k-1\right)\right] \\ &\quad + \tilde{u}_{eq}(k-1). \end{aligned} \tag{2.66}$$

Taking into account the control constrain (2.44), we define $u(k)$ [44], as:

$$u(k) = \begin{cases} \tilde{u}_{eq}(k) & \text{for } \|\tilde{u}_{eq}(k)\| \leq u_0 \\ u_0 \frac{u_s(k)}{\|u_s(k)\|} & \text{for } \|\tilde{u}_{eq}(k)\| > u_0 \end{cases} \tag{2.67}$$

To proceed with the stability analysis, we establish that the closed-loop system ((2.60) and (2.67)) motion over the manifold $s_D(k+1) = 0$ is stable.

First, to reveal the structure of the control $u_s(k)$ (2.63) and system (2.60) let us represent them, respectively, by imposing the term $\left(s_D(k) - \chi^r(k) + \chi_d^r(k) = 0\right)$:

$$\begin{aligned} u_s(k) &= -\left[\bar{W}_r\right]^{-1}\left[s_D(k) - \chi^r(k) + \chi_d^r(k)\right. \\ &\quad \left. + \tilde{f}_r\left(\chi(k), k\right)\right] \\ &= -\left[\bar{W}_r\right]^{-1}\left[s_D(k) + f_s\left(\chi(k), k\right)\right] \end{aligned}$$

and

$$\begin{aligned} s_D(k+1) &= s_D(k) - \chi^r(k) + \chi_d^r(k) \\ &\quad + \tilde{f}_r\left(\chi(k), k\right) + \bar{W}_r u(k) + \tilde{\Delta}_r(k) \\ &= s_D(k) + f_s\left(\chi(k), k\right) + \bar{W}_r u(k) + \tilde{\Delta}_r(k) \end{aligned}$$

where

$$f_s\left(\chi(k), k\right) = -\chi^r(k) + \chi_d^r(k) + \tilde{f}_r\left(\chi(k), k\right). \tag{2.68}$$

To this end we introduce the following assumption.

Assumption 3 *The maximum value u_0 of the control is such that*

$$u_0 > \delta_r, \quad \delta_r = \left\| [\bar{W}_r]^{-1} \right\| \left(\|f_s(k)\| + \left\| \tilde{\Delta}_r(k) \right\| \right). \tag{2.69}$$

This assumption implies that the control resource is enough to guarantee the achievement of control requirements. Now we are ready to formulate the following result.

Proposition 3 *If the Assumption 3 (2.69) is fulfilled, then the control law (2.67) with (2.66) and (2.63) ensures for system (2.45) the sliding manifold $s_D(k) = 0$ is stable and there is a time k_1 such that*

$$\|s_D(k)\| \leq \xi_r \quad for \quad k \geq k_1,$$

with $\xi_r = \left\| \tilde{\Delta}_r(k) - \tilde{\Delta}_r(k-1) \right\|$.

Proof. For the case $\|\tilde{u}_{eq}(k)\| \leq u_0$, $\tilde{u}_{eq}(k)$ is applied, yielding motion in the neighborhood of the sliding manifold $\|s_D(k)\| \leq \xi_r$ at time $k_1 + 1$.

For the case $\|\tilde{u}_{eq}(k)\| > u_0$, the proposed control strategy is $u(k) = u_0 \frac{u_s(k)}{\|u_s(k)\|}$, and the closed-loop system becomes

$$\begin{aligned} s_D(k+1) &= s_D(k) + f_s(\chi(k), k) \\ &\quad + \bar{W}_r u_0 \frac{u_s(k)}{\|u_s(k)\|} + \tilde{\Delta}_r(k) \\ &= [s_D(k) - f_s(\chi(k), k)] \left(1 - \frac{u_0}{\|u_s(k)\|} \right) \\ &\quad + \tilde{\Delta}_r(k). \end{aligned} \tag{2.70}$$

The rate of change of a Lyapunov function candidate function $V(k) = \|s_D(k)\|$ along the motion of the system (2.70) is given by

$$\begin{aligned} \Delta V(k) &= \|s_D(k+1)\| - \|s_D(k)\| \\ &\leq \|s_D(k) + f_s(k)\| \left(1 - \frac{u_0}{\|u_s(k)\|} \right) \\ &\quad + \left\| \tilde{\Delta}_r(k) \right\| - \|s_D(k)\| \\ &\leq \left(\|s_D(k) + f_s(k)\| - \frac{u_0}{\left\| [\bar{W}_r']^{-1} \right\|} + \left\| \tilde{\Delta}_r(k) \right\| \right) \\ &\quad - \|s_D(k)\|. \end{aligned}$$

Applying Assumption 3 (2.69) yields

$$\left(\|s_D(k) + f_s(k)\| - \frac{u_0}{\left\| \left[\bar{W}_r\right]^{-1} \right\|} + \left\| \tilde{\Delta}_r(k) \right\| \right)$$
$$\leq \|s_D(k)\| + \delta_r - \frac{u_0}{\left\| \left[\bar{W}_r\right]^{-1} \right\|} < \|s_D(k)\| .$$

Therefore $\Delta V < 0$, which implies $\|s_D(k)\|$ decreases monotonically. Substituting $\tilde{f}_r\left(\chi(k-1), k-1\right) = s_D(k) - \bar{W}_r u(k-1) - \tilde{\Delta}_r(k-1)$ and then $\tilde{f}_r\left(\chi(k), k\right) = s_D\left(k\right) + f_s\left(\chi(k), k\right)$ (2.68) in equation (2.66) we have

$$\begin{aligned}
\tilde{u}_{eq}(k) &= -\left[\bar{W}_r\right]^{-1}\left[s_D(k) + \tilde{f}_r\left(\chi(k), k\right)\right] \\
&\quad -\left[\bar{W}_r\right]^{-1}\left[-\tilde{f}_r\left(\chi(k-1), k-1\right)\right] \\
&\quad +u(k-1) \\
&= -\left[\bar{W}_r\right]^{-1}\left[\tilde{f}_r\left(\chi(k), k\right) + \tilde{\Delta}_r(k-1)\right] \\
&= -\left[\bar{W}_r\right]^{-1}\left[s_D\left(k\right) + f_s\left(\chi(k), k\right) + \tilde{\Delta}_r(k-1)\right]
\end{aligned}$$

Now using (2.69) results in

$$\|\tilde{u}_{eq}(k)\| \leq \left\| \left[\bar{W}_r\right]^{-1} \right\| \|s_D(k)\| + \delta_r.$$

Therefore there will be a time k_1 such that $\|\tilde{u}_{eq}(k)\| \leq u_0$, for $k \geq k_1$. At that time, the equivalent control $u(k) = \tilde{u}_{eq}(k)$ (2.66) is applied, yielding the motion in the neighborhood of the sliding manifold $\|s_D(k)\| \leq \xi_r$ at time $k_1 + 1$. ■

2.5.3 Sliding Mode Dynamics

The above results implies that the closed-loop system motion remains within an $O(T_s)$ boundary layer [46] of $s_D = 0$, with T_s as the sampling period. This motion is governed by the following reduced order system (Sliding Mode Equation (SME)) derived from (2.51):

$$\varepsilon(k+1) = A_s\varepsilon(k) + \tilde{\Delta}(k) \tag{2.71}$$

where $\varepsilon = \left[\varepsilon^{1T}\ \varepsilon^{2T} \ldots \varepsilon^{(r-1)T}\right]^T$, $A_s = \mathbf{K} + \tilde{W}$.

$$\begin{aligned}
\mathbf{K} &= diag\left\{\mathbf{K}_1, \cdots, \mathbf{K}_{r-1}\right\}. \\
\bar{W} &= subdiag\left\{\bar{W}_1, \cdots, \bar{W}_{r-2}, 0\right\}. \\
\tilde{\Delta}(k) &= [\tilde{\Delta}_1(k+1), \cdots, \tilde{\Delta}_{r-2}(k+1), \\
&\quad \bar{\Delta}_{r-1}(k+1)]^T \\
\bar{\Delta}_{r-1}(k+1) &= \bar{W}_{r-1}(\tilde{\Delta}_r(k) - \tilde{\Delta}_{r-1}(k)) + \tilde{\Delta}_{r-1}(k).
\end{aligned}$$

To analyze the stability of the linear perturbed system (2.71) consider the Lyapunov function

$$V(k) = \varepsilon^T(k) P \varepsilon(k) \tag{2.72}$$

with a P positive definite solution of the following Lyapunov equation:

$$A_s^T P \mathbf{K}_s - P = -Q, \; Q > 0, \; Q = Q^T \tag{2.73}$$

Proposition 4 *If for some $\gamma > 0$ the following condition:*

$$\alpha_{\min}(Q+P) - \gamma \left\| A_s^T P \right\| - \gamma^2 \alpha_{\max}(P) > 0 \tag{2.74}$$

holds, then there is $k_2 > k_1$ such that a solution of the SME (2.71) is uniformly ultimately bounded by

$$\|\varepsilon(k)\| \leq \delta, \; \forall k > k_2, \; \delta = \frac{1}{\gamma} \|\Delta(k)\|. \tag{2.75}$$

Proof. Calculating the difference of the Lypunov function candidate (2.72) along the trajectories of the system (2.71) yields

$$\begin{aligned}
\Delta V(k) &= V(k+1) - V(k) \\
&= \varepsilon^T(k+1) P \varepsilon(k+1) - \varepsilon^T(k) P \varepsilon(k) \\
&= \left(\varepsilon^T A_s^T(k) + \Delta^T(k)\right) P \left(A_s \varepsilon(k) + \Delta(k)\right) \\
&\quad -\varepsilon^T(k) P \varepsilon(k) \\
&= \varepsilon^T(k) A_s^T P A_s \varepsilon(k) + 2\varepsilon^T(k) A_s^T P \Delta(k) \\
&\quad +\Delta^T(k) P \Delta(k) - \varepsilon^T(k) P \varepsilon(k).
\end{aligned} \tag{2.76}$$

Then using (2.73), we have

$$\begin{aligned}\Delta V(k) &= -\varepsilon^T(k)(Q+P)\varepsilon(k) \\ &\quad +\varepsilon^T(k)D\Delta(k)+\Delta^T(k)P\Delta(k) \\ &\leq -\alpha_{min}(Q+P)\left\|\varepsilon(k)\right\|^2 \\ &\quad +\left\|A_s^TP\right\|\left\|\varepsilon(k)\right\|\left\|\Delta(k)\right\| \\ &\quad +\alpha_{\max}(P)\left\|\Delta(k)\right\|^2\end{aligned}$$

Now, introducing a bound

$$\left\|\varepsilon(k)\right\| > \frac{1}{\gamma}\left\|\Delta(k)\right\|, \quad \gamma > 0 \tag{2.77}$$

results in

$$\begin{aligned}\Delta V(k) &\leq -\alpha_{\min}(Q+P)\left\|\varepsilon(k)\right\|^2 \\ &\quad +\left[\gamma\left\|A_s^TP\right\|+\gamma^2\alpha_{\max}(P)\right]\left\|\varepsilon(k)\right\|^2\end{aligned}$$

Now, it is evidently that if the condition (2.74), namely

$$\alpha_{\min}(Q+P)-\gamma\left\|A_s^TP\right\|-\gamma^2\alpha_{\max}(P)>0$$

is fulfilled then $\Delta V(k)$ is negative outside the region (2.77). Hence there exist a time instant k_2 such that the solution of system (2.71) enters the region (2.75) and will remain in it for all $\forall k > k_2$. ■

Results obtained in Propositions *(2)-(4)*, can be formulated in the following theorem.

Theorem 4 *If the Assumption 3 (2.69) and the condition (2.74) are satisfied then a solution of the closed-loop system (2.45) using (2.67) and the tracking error (2.49) are uniformly ultimately bounded.*

Chapter 3

Experimental Prototype

This chapter describes the prototype in which the experimental tests are performed. The schematic representation of the control prototype is presented in Fig. 3.1, and the prototype parts description are included below.

3.1 DC Machine

The DC motor used for this application is a separate winding excitation one; it means that the field and armature windings are excited from separate sources. This motor has two field coils which are connected in series (shunt wound connection). The connection made in this application with the DC motor is shown in Fig. 3.2. One of the advantages of this connection is that is possible to control the angular speed or the electromagnetic torque without changing the connection. The terminals ($F1$ and $F4$) and ($A1$ and $A2$) are connected to the IGBT teaching system, as it is shown in Fig. 3.12. Fig. 3.3 shows the DC motor used for the present work. The main characteristics of this motor are:

- 1750 RPM.
- Armature voltage 90 V
- Field voltage 100 V
- Armature current 3 A.
- Field current 0.3 A.
- Electrical power 0.25 HP

For more information of the DC motor consult [3].

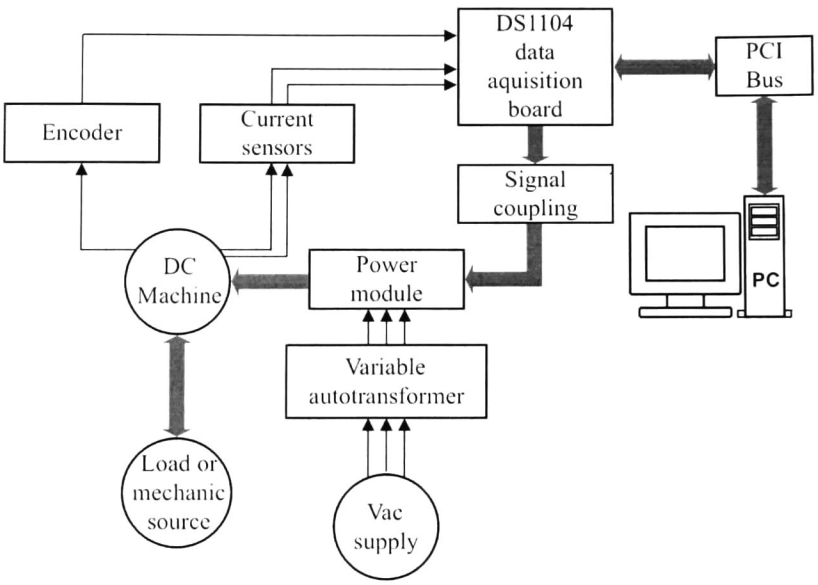

Figure 3.1: Schematic representation of the control prototype.

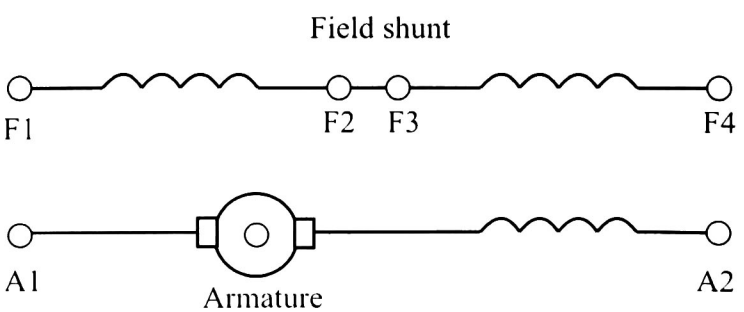

Figure 3.2: DC motor with shunt wound connection.

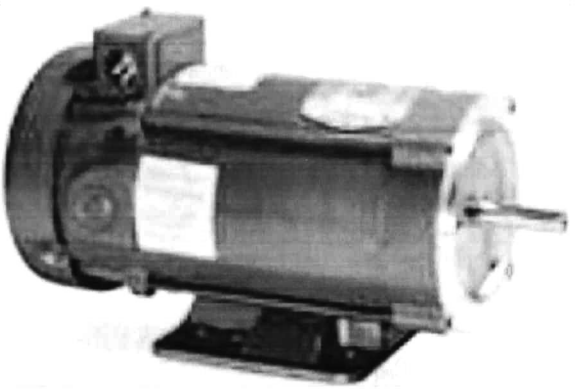

Figure 3.3: DC motor.

3.2 Incremental Encoder

The incremental encoder is used for the measurement of the angular speed. The model used for this application is H25E-SS-1024. The main characteristics are described below.

3.2.1 Description

- It produces a series of square waves as it rotates (channels A and B) which are offsets each other by 1/4 of a cycle (90 degrees). This type of signal is referred to as quadrature and allows the user to determine not only the speed but its direction as well.
- Resolution of 1024 cycles per turn.
- Supply voltage: 5 to 28 $Volts$.
- Frequency response: 1000 khz.
- Maximum RPM measurement: 6,000.

The incremental encoder is connected directly to the incremental encoder interface of the data acquisition board DS1104, as is shown in Fig. 3.4. Fig. 3.5 presents the incremental encoder used for the present project. For more details of the incremental encoder, consult [4]. Fig. 3.6 displays the incremental encoder coupled with the DC motor.

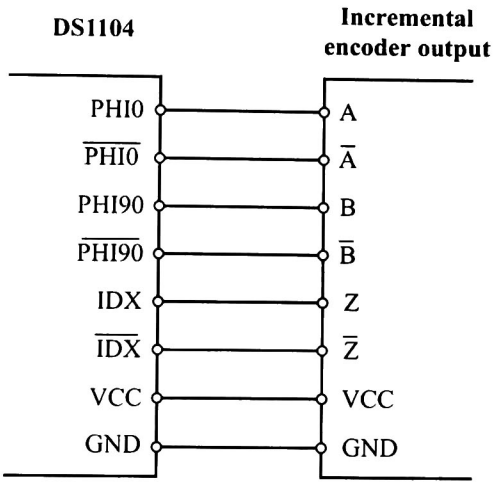

Figure 3.4: Connection between the Incremental encoder and the DS1104 data acquisition board.

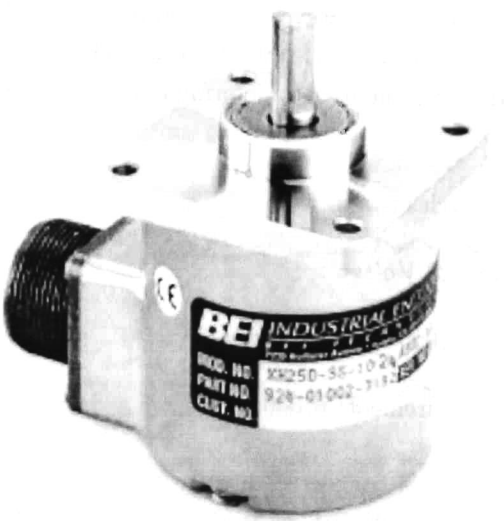

Figure 3.5: Incremental encoder H25E-SS-1024.

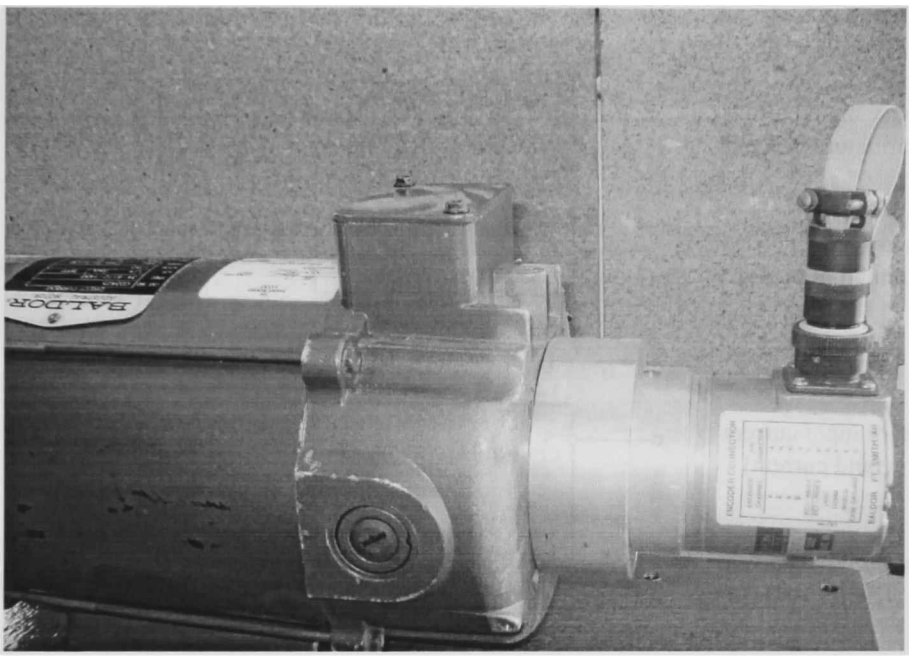

Figure 3.6: Incremental encoder coupled with the DC motor.

3.3 Current Sensors

The armature and field currents are measured with the LEM HX-10P current sensor. In this application, we use sensors whose characteristics are:

- Galvanic isolation between primary and secondary circuit.
- Hall effect measuring principle.
- Isolation voltage 3000 V
- Low power consumption.
- Power supply from ± 12 V to ± 15 V
- Output voltage ± 4 V with a measurement of ± 10 A.

For DC motor drives, the application of this kind of sensors, present the following advantages:

- Low insertion losses.

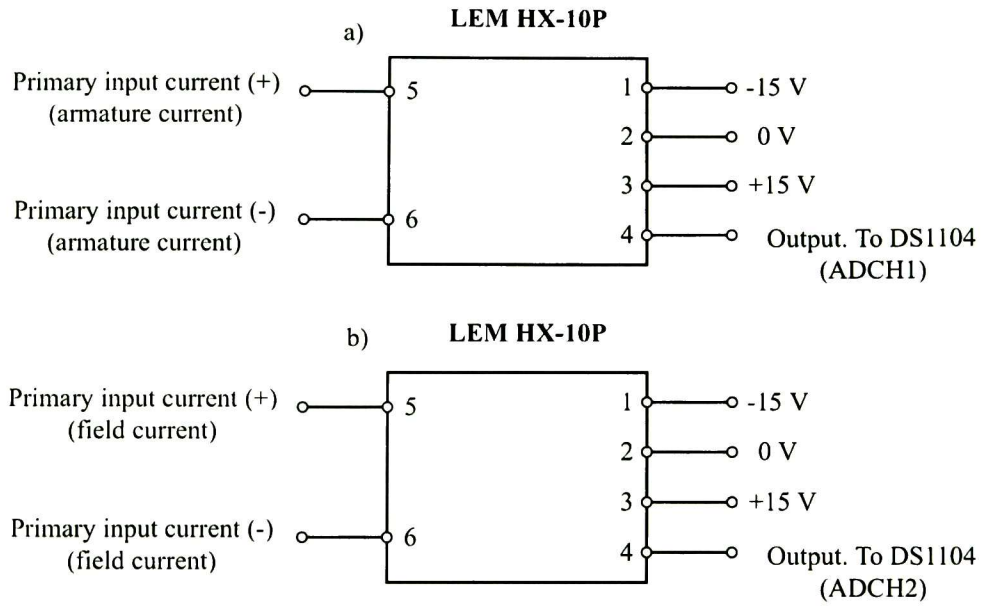

Figure 3.7: Connection of the LEM HX-10P current sensors with the DS1104.

- Easy to install with automatic handling system.
- Small size and space saving.
- Wide current ratings range.
- High immunity to external interference.

The output of the current sensors are connected directly to two inputs of the A/D of the DS1104 data acquisition board (ADCH1 and ADCH2). Fig. 3.7 displays the connection of the LEM HX-10P current sensors with the DS1104 data acquisition board and the DC motor, for measuring: a) the armature current and b) the field current. For more information about the current sensors, consult [21]. Fig. 3.8 shows the LEM HX-10P.

3.4 DS1104 Data Acquisition Board

The data acquisiton stand alone board used for real-time implementation is the DS1104 manufactured by dSPACE. The main characteristics of this board are:

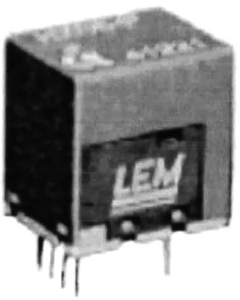

Figure 3.8: Current sensor LEM HX-10P.

- Cost-effective system for controller development
- Single-board PCI hardware for use in PCs
- Set of Intelligent I/O on-board
- Incremental encoder interface
- Serial Interface (UART)
- Application areas. The real-time hardware based on PowerPC technology and its set of I/O interfaces make this board an ideal solution for developing controllers in various fields, such as drivers, robotics, aerospace and automotive.
- Key benefits. It upgrades the PC to a powerful development system for rapid control prototyping. Real-Time Interface provides Simulink blocks for graphical configuration of analog to digital converters (A/D), digital to analog converters (D/A), digital I/O lines, incremental encoder interface and PWM generation, for example. The board can be installed in virtually any PC with a free 5 V PCI slot.
- Using Real-Time Interface (RTI). With RTI, it can easily run the function models on the DS1104 Controller board. It is possible to configure all I/O graphically by dragging RTI blocks and to reduce implementation time to a minimum.

Fig. 3.9 presents the DS1104 data acquisition board and Fig. 3.10 portraits a view of the DS1104 board connected to the PCI bus in the PC. For more information about the data acquisition board DS1104, consult [9].

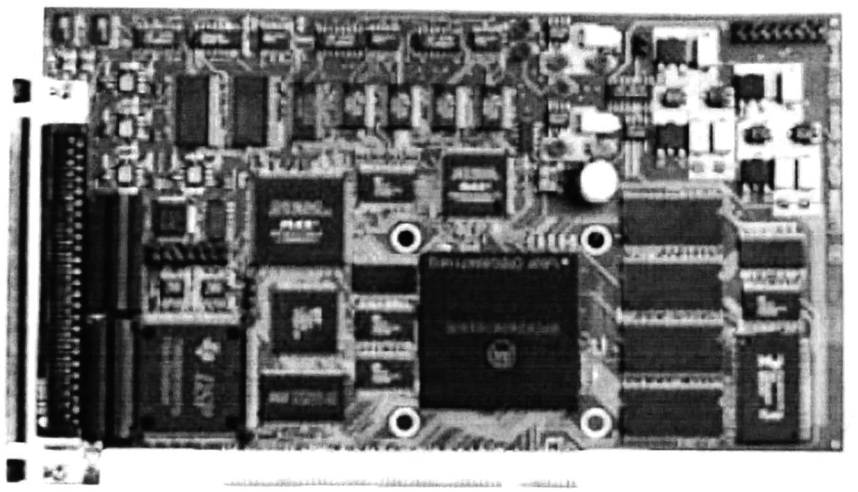

Figure 3.9: DS1104 data acquisition board.

3.5 Signal Coupling

The digital outputs of the DS1104 data acquisition board are TTL volts (+5 V for 1 and 0 V for 0). Meanwhile, the logic levels of the IGBT system are CMOS (+15 V for 1 and 0 V for 0). Therefore, it is necessary to couple the TTL to CMOS voltages, particularly the $SPWM3$ (phase 1, 2 and 3), $IO1$, $IO2$ and $IO5$ outputs of the data acquisition board.

This coupling must:

- to have sufficient capacity of response for not to retard the signals of the $SPWM3$.
- to work at the frequency of the IGBT system module that is 20 Khz.
- to have high immunity to external interference.
- to have protections to minimize safety hazards.
- to have independent power supply.

Fig. 3.11 presents the TTL-CMOS coupling as well as connections between the DS1104 data acquisition board with the IGBT system. The TTL-CMOS coupling consists of the following parts:

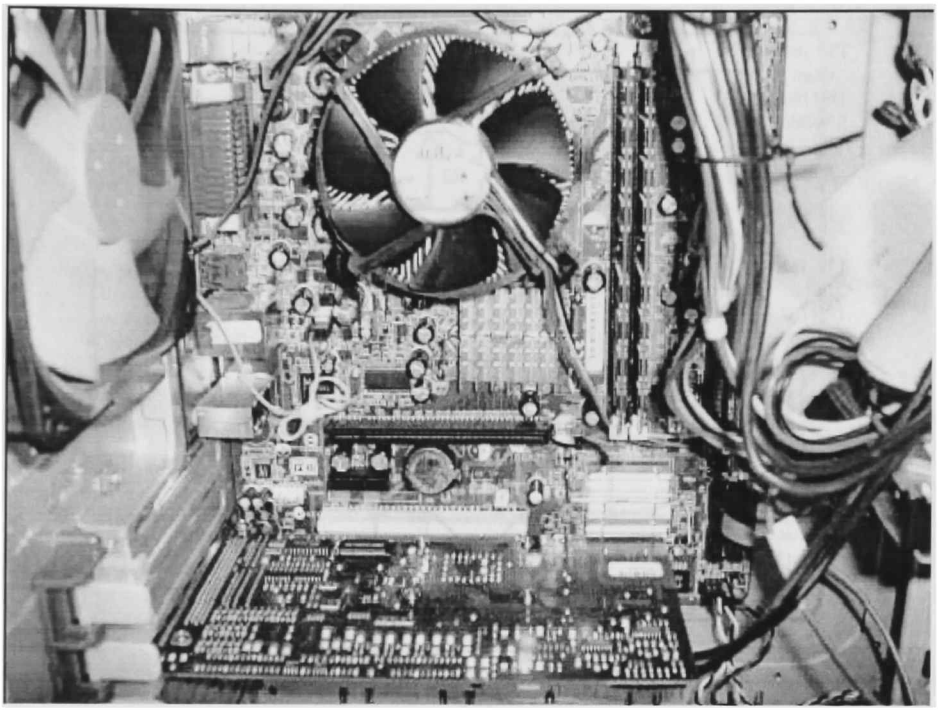

Figure 3.10: View of the DS1104 board connected to the PCI bus in the PC.

- Transistor 2N3904 NPN (Q1).
- Diode 1N904 (D1).
- Integrated Circuit 74LS04 Inverter gate (IC1).
- Integrated Circuit MM74HC4049 Inverter gate (IC2).
- Resistances of 4.7 $K\Omega$ (R1 and R2).
- Resistances of 3.9 $K\Omega$ (R3).

3.6 Power Module

The power module is constituted by the IGBT teaching system, which is a three-phase rectifier plus inverter with brake chopper. It is manufactured by SEMIKRON and is used specially for motors. For this dissertation, it is used as a rectifier and the AC voltage is supplied from the three phase variable autotransformer. The signal rectified is

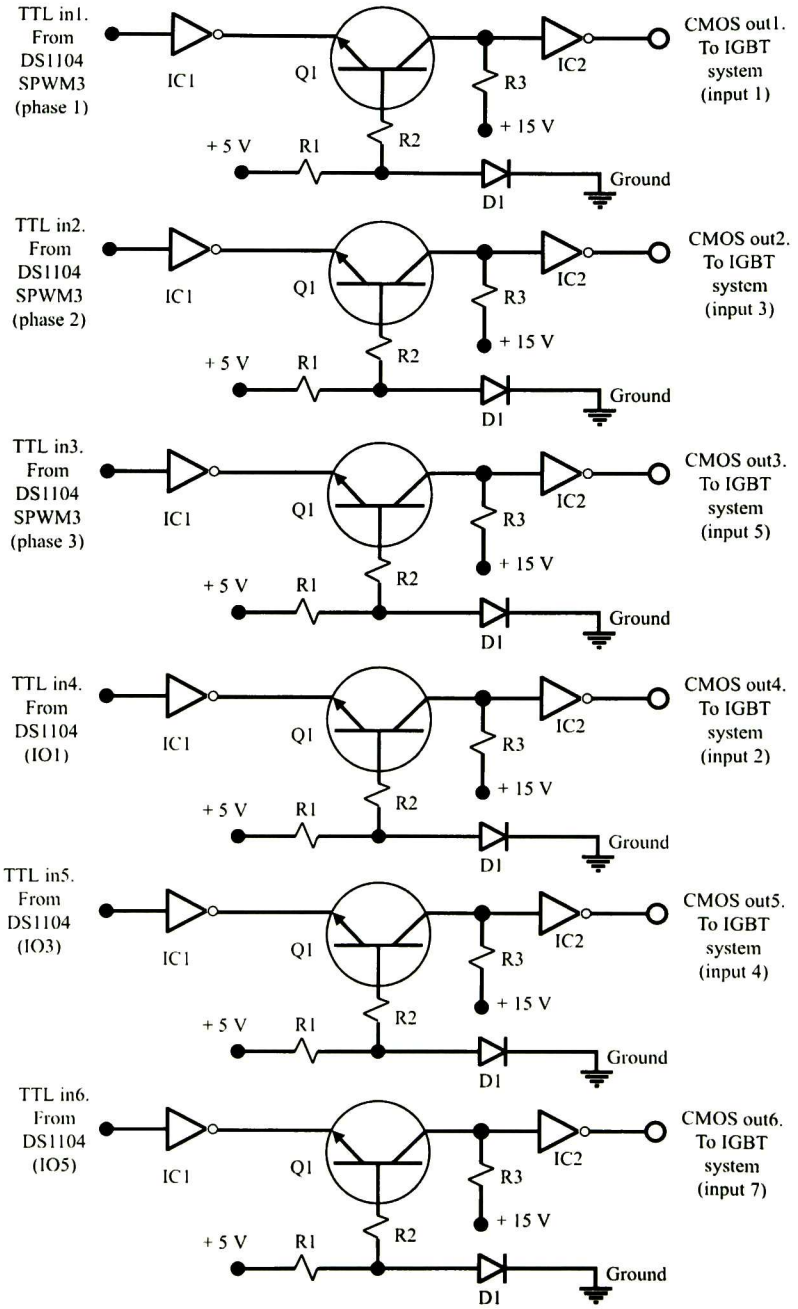

Figure 3.11: TTL-CMOS coupling.

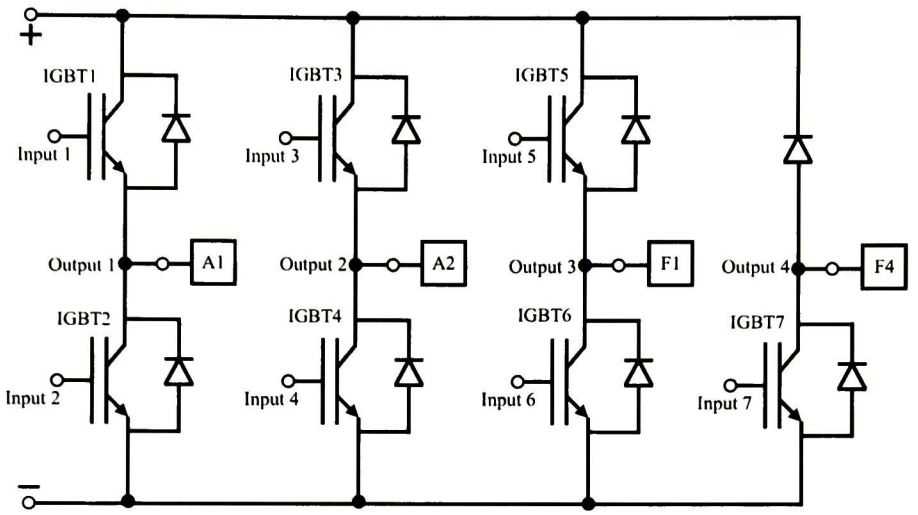

Figure 3.12: Connection between the IGBT system and the DC motor.

used for providing the DC voltage to the DC motor. The IGBT system has seven inputs (inputs 1-7) which are excited from the data acquisition board (through TTL-CMOS coupling), as shown in Fig. 3.12. Only inputs 1 and 4 of the IGBT system are excited when the armature winding is polarized for clock wise rotation, and only inputs 2 and 3 are polarized when the armature is polarized for counter clock wise rotation. Inputs 5 and 7 are excited to create the electromagnetic field required for the DC Motor. Inputs 1, 3 and 5 are excited from the $SPWM3$ (phase 1, 2 and 3 respectively) outputs of the data acquisition board, for giving the appropriate DC voltage to the armature winding (input 1 or 3) and to the field winding (input 5). Terminals $SPWM3$ (phase 1, 2 and 3) of the data acquisition board provide the pulse wide modulation (PWM) for creating the appropriate voltage. Connection of the DC motor and the IGBT system is shown in Fig. 3.12; output terminals 1 and 2 are connected to the armature winding terminals $A1$ and $A2$, respectively and output terminals 3 and 4 are connected to the field winding $F1$ and $F4$, respectively. Fig. 3.13 shows the IGBT Teaching System. For more information about the IGBT system, consult [38].

3.7 Computer Station

A PC is used for supervision. The software of the DS1104 data acquisition board is compatible with the software of MATLAB and Simulink of Mathworks. Fig. 3.14 shows

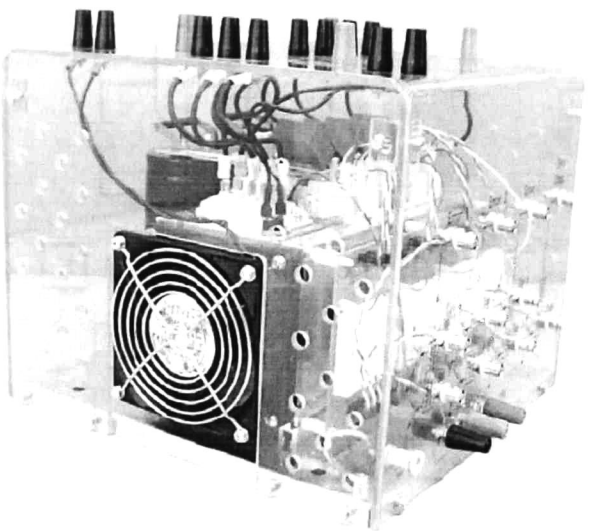

Figure 3.13: IGBT Teaching system.

the application of the DS1104 board software directly from Simulink. Fig. 3.15 presents a desktop interface for the DS1104 board, to display the real-time experiments.

3.8 Variable Autotransformer

This autotransformer is used for providing the AC voltage to the IGBT system. The variable autotransformer connection is presented in Fig. 3.16 and a view of it is shown in Fig. 3.17. The main characteristics of the variable autotransformer are:

- Input 240 Vac, 60 hz.
- Output 280 Vac, 15 A, 7.26 KVA.

For more information about the variable autotransformer, consult [43].

3.9 Mechanical Load

The mechanical load is constituted by a mechanical coupling between the DC motor with an induction motor, as is shown in Fig. 3.18. This mechanical coupling is for providing an inertial load to the DC motor.

A view of the complete prototype is presented in Fig. 3.18

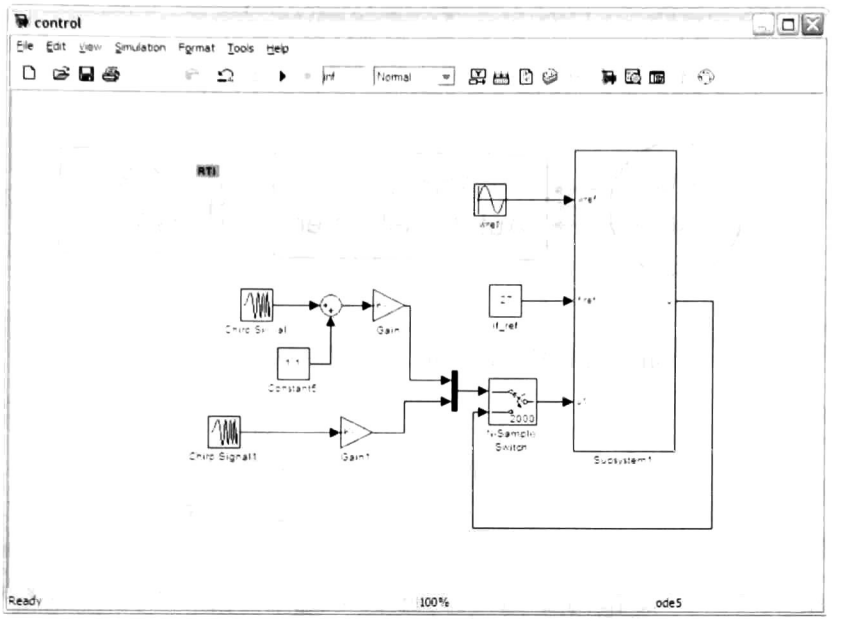

Figure 3.14: Simulink program to be downloaded to the DS1104 board.

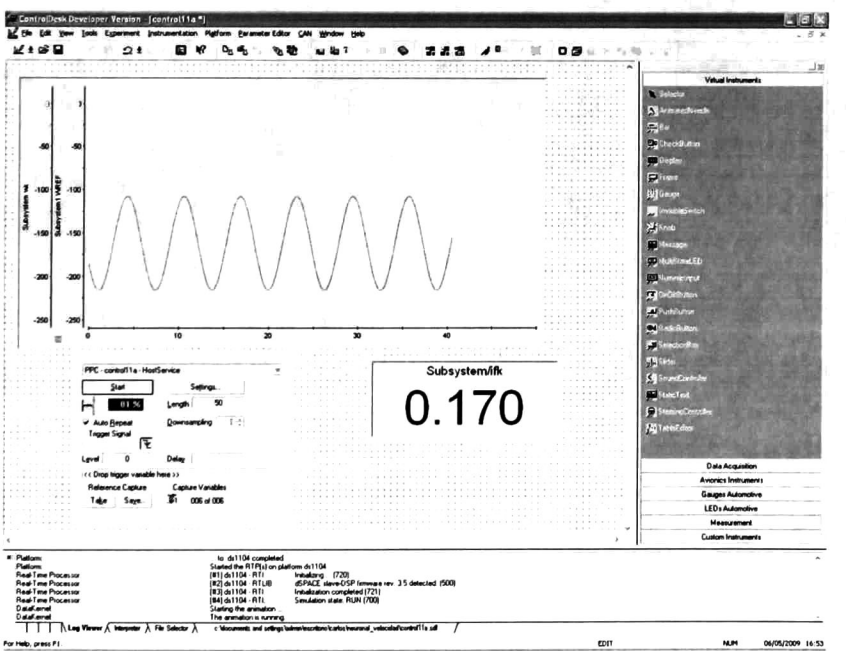

Figure 3.15: Desktop interface for the DS1104 board.

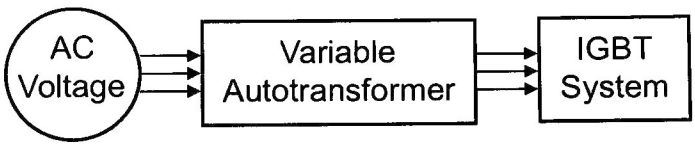

Figure 3.16: Variable autotransformer connection.

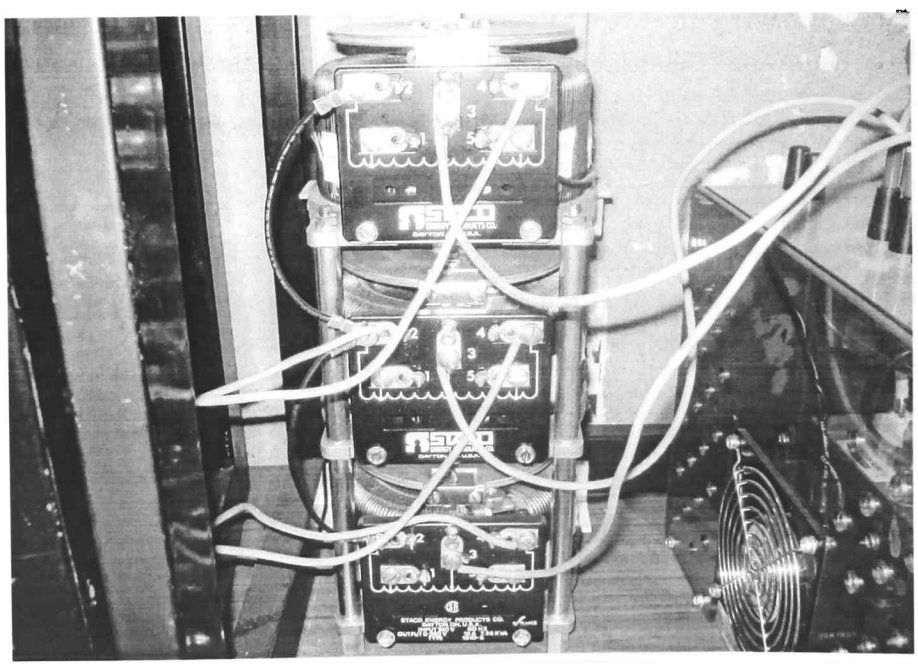

Figure 3.17: View of the variable autotransformer.

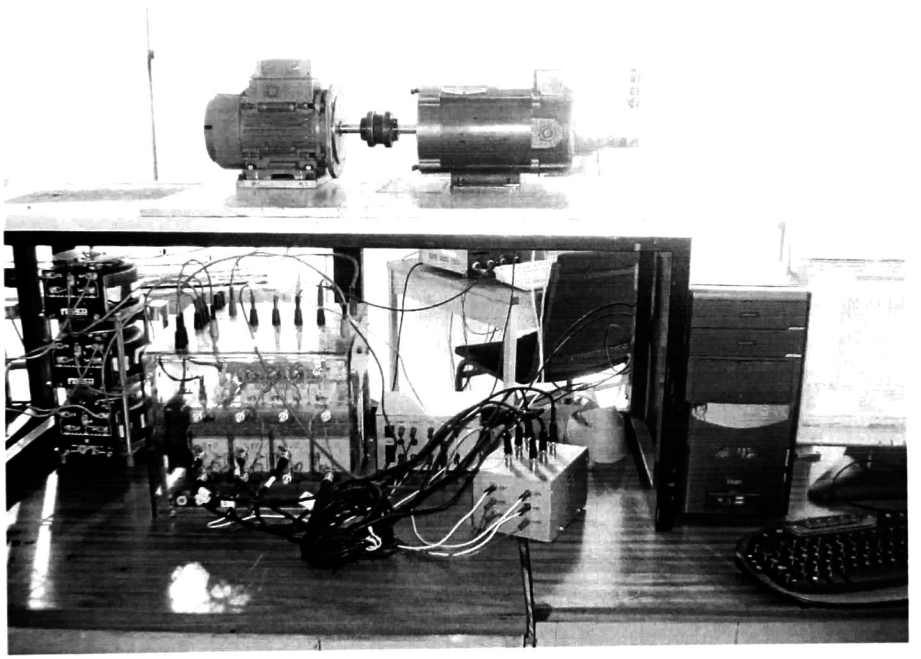

Figure 3.18: Complete prototype.

Chapter 4

Neural DC Motor Identification and Control

In this chapter, we test the applicability of the identification and control scheme proposed in Chapter 2 in order to illustrate identification of the plant model and the control of the angular speed and the electromagnetic torque for a DC motor with separate winding excitation, whose model is presented in section 2.2. Additionally, simulations and real-time implementation results are presented.

4.1 Neural Network Identification

Based on the DC motor discrete-time model (2.19), the following RHONN is proposed:

$$\begin{aligned} x_1(k+1) &= w_{11}(k)S\left(\omega_m(k)\right) + \bar{w}_1 i_a(k) \\ x_2(k+1) &= w_{21}(k)S\left(\omega_m(k)\right)S\left(i_f(k)\right) + w_{22}(k)S\left(i_a(k)\right) \\ &\quad + w_{23}(k)S\left(i_f(k)\right) + \bar{w}_2 u_1(k) \\ x_3(k+1) &= w_{31}(k)S\left(i_f(k)\right) + \bar{w}_3 u_2(k) \end{aligned} \tag{4.1}$$

where $x_i(k)$, $i = 1, 2, 3$ is the state of the ith neuron; $x_1(k)$ is the estimate of the angular speed $\omega_m(k)$; $x_2(k)$ is the estimate of the armature current $i_a(k)$; $x_3(k)$ is the estimate of the field current $i_f(k)$; $u_1(k) = u_a(k)$ is the voltage applied to the armature winding; $u_2(k) = u_f(k)$ is the votage applied to the filed winding; $w_{11}(k)$, $w_{21}(k)$, $w_{22}(k)$, $w_{23}(k)$ and $w_{31}(k)$ are the respective on-line adapted weights; the weights $\bar{w}_1$, $\bar{w}_2$ and $\bar{w}_3$ are constant positive parameters, which are selected in order to minimize the identification error and are called the controllability weights [10]; and $S(\bullet)$ is defined in (2.23). We propose this

structure because it is triangular, similar to (2.45), and allows the application of the BC technique as explained in Chapter 2. Moreover, this structure uses a reduced number of weights to obtain an adequate identification.

The training of the proposed RHONN (4.1) uses the EKF algorithm (2.31)-(2.33) and it is performed on-line using a series-parallel configuration ([15], [35]). All the NN states are initialized in a random way as well as the weights vectors. It is important to note that the initial conditions of the NN are completely different from the plant initial conditions. The weights constant values are: $\bar{w}_1 = 0.0150$, $\bar{w}_2 = 191.93$ and $\bar{w}_3 = 0.0001$. The covariance matrices are initialized as diagonals, which their corresponding values are: $P_1(0) = P_3(0) = 1 \times 10^8$, $P_2(0) = 1 \times 10^4$, $Q_1(0) = 1 \times 10^5$, $Q_2(0) = 1 \times 10^8$, $Q_3 = 1 \times 10^3$, $R_1(0) = R_3(0) = 1 \times 10^4$ and $R_2(0) = 5 \times 10^4$

4.2 Angular Speed Control

The goal is to define a control law to achieve angular speed control. Considering $\chi_1(k) = \omega_m(k)$, $\chi_2(k) = i_a(k)$, $\chi_3(k) = i_f(k)$, $z_1(\chi_1(k)) = S(w_m(k))$, $\hat{z}_2(\chi(k)) = [S(\omega_m(k))\, S(i_f(k)) \quad S(i_a(k)) \quad S(i_f(k))]^T$, $z_3(\chi_3(k)) = S(i_f(k))$, $w_1(k) = w_{11}(k)$, $w_3(k) = w_{31}(k)$, $\hat{w}_2(k) = [w_{21}(k) \quad w_{22}(k) \quad w_{23}(k)]^T$ then system (4.1) can be represented as the BC form consisting of two blocks (similar to (2.45)):

$$\{x_1(k+1) = w_1(k) z_1(\chi_1(k)) + \bar{w}_1 \chi_2(k) \tag{4.2}$$

$$\begin{cases} x_2(k+1) = \hat{w}_2^T(k) \hat{z}_2(\chi(k)) + \bar{w}_2 u_1(k) \\ x_3(k+1) = w_3(k) z_3(\chi_3(k)) + \bar{w}_3 u_2(k) \end{cases} \tag{4.3}$$

where $x^1 = x_1$, $x^2 = [x_2 \quad x_3]^T$. $n_1 = 1$, $n_2 = 2$ and $r = 2$.

At the first step, let us define $\chi_{1d}(k)$ and $\chi_{3d}(k)$ as the speed and field current desired trajectories, respectively. Defining the speed error

$$\begin{aligned} \varepsilon_1(k) &= \chi^1(k) - \chi_d^1(k) \\ &= x^1(k) - \chi_d^1(k) + \Delta_1(k) \end{aligned}$$

and applying the block control technique for the first block (4.2), we have

$$\begin{aligned} \varepsilon_1(k+1) &= \tilde{f}_1(\chi_1(k), k) + \bar{w}_1 \chi_2(k) + \tilde{\Delta}_1(k) \\ &= \mathbf{k}_1 \varepsilon_1(k) + \tilde{\Delta}_1(k) \end{aligned} \tag{4.4}$$

where $\tilde{f}_1(\chi_1(k),k) = w_1(k)z_1(\chi_1(k)) - \chi_{1d}(k+1)$, $|\mathbf{k}_1| < 1$, $\mathbf{k}_1\varepsilon_1(k)$ is the desired dynamics for $\varepsilon_1(k)$ and $\tilde{\Delta}_1(k) = \Delta_1(k+1)$. Then the desired value $\chi_{2d}(k)$ for $\chi_2(k)$ is calculated from (4.4) as

$$\chi_{2d}(k) = [\bar{w}_1]^{-1}\left[-\tilde{f}_1(\chi_1(k),k) + \mathbf{k}_1\varepsilon_1(k)\right]$$

At the second step, given $\chi_{2d}(k)$ and $\chi_{3d}(k)$ as the desired values for the armature $\chi_2(k)$ and field $\chi_3(k)$ currents, respectively, a second error vector $\varepsilon^2(k)$ is defined as

$$\begin{aligned}\varepsilon^2(k) &\equiv \begin{bmatrix}\varepsilon_2(k)\\ \varepsilon_3(k)\end{bmatrix} = \begin{bmatrix}\chi_2(k) - \chi_{2d}(k)\\ \chi_3(k) - \chi_{3d}(k)\end{bmatrix}\\ &= \begin{bmatrix}x_2(k) - \chi_{2d}(k) + \Delta_2(k)\\ x_3(k) - \chi_{3d}(k) + \Delta_3(k)\end{bmatrix}\end{aligned}$$

Thus, the system (4.2)-(4.3) in the new variables ε_1, ε_2 and ε_3 is presented of the form

$$\left\{\varepsilon_1(k+1) = \mathbf{k}_1\varepsilon_1(k) + \bar{w}_1\varepsilon_2(k) + \tilde{\Delta}_1(k)\right. \tag{4.5}$$

$$\left\{\begin{array}{l}\varepsilon_2(k+1) = \tilde{f}_2(\chi(k),k) + \bar{w}_2u_1(k) + \tilde{\Delta}_2(k)\\ \varepsilon_3(k+1) = \tilde{f}_3(\chi_3(k),k) + \bar{w}_3u_2(k) + \tilde{\Delta}_3(k)\end{array}\right. \tag{4.6}$$

where $\chi(k) = [\chi_1(k), \chi_2(k), \chi_3(k)]^T$, and $\tilde{\Delta}_i(k) = \Delta_i(k+1)$, $i = 2,3$. Selecting the sliding variables as $s_2(k) = \varepsilon_2(k)$, and $s_3(k) = \varepsilon_3(k)$ and taking into account the respective bounds $\|u_1\| \leq u_{01}$ and $\|u_2\| \leq u_{02}$ for the armature and field voltages, respectively, the following control is proposed (2.67):

$$u_1(k) = \left\{\begin{array}{ll}\tilde{u}_{1eq}(k) & \text{for } \|\tilde{u}_{1eq}(k)\| \leq u_{01}\\ u_{01}\frac{u_{1s}}{\|u_{1s}(k)\|} & \text{for } \|\tilde{u}_{1eq}(k)\| > u_{01}\end{array}\right. \tag{4.7}$$

$$\begin{aligned}\tilde{u}_{1eq}(k) &= -[\bar{w}_2]^{-1}\left[s_2(k) + \tilde{f}_2(\chi(k),k)\right]\\ &\quad + [\bar{w}_2]^{-1}\left[\tilde{f}_2(\chi(k-1),k-1)\right] + \tilde{u}_{1eq}(k-1)\\ u_{1s}(k) &= -[\bar{w}_2]^{-1}\tilde{f}_2(\chi(k),k)\end{aligned}$$

and

$$u_2(k) = \left\{\begin{array}{ll}\tilde{u}_{2eq}(k) & \text{for } \|\tilde{u}_{2eq}(k)\| \leq u_{02}\\ u_{02}\frac{u_{2s}}{\|u_{2s}(k)\|} & \text{for } \|\tilde{u}_{2eq}(k)\| > u_{02}\end{array}\right. \tag{4.8}$$

with

$$\begin{aligned}\tilde{u}_{2eq}(k) &= -\left[\bar{w}_3\right]^{-1}\left[s_3(k)+\tilde{f}_3\left(\chi_3(k),k\right)\right] \\ &\quad +\left[\bar{w}_3\right]^{-1}\left[\tilde{f}_3\left(\chi_3(k-1),k-1\right)\right]+\tilde{u}_{2eq}(k-1) \\ u_{2s}(k) &= -\left[\bar{w}_3\right]^{-1}\tilde{f}_3\left(\chi_3(k),k\right).\end{aligned}$$

Considering Proposition 3, under the condition

$$\begin{aligned}\bar{w}_2 u_{10} &> \delta_1, \quad \delta_1 = \left(\|f_{2s}(k)\| + \left\|\tilde{\Delta}_2(k)\right\|\right) \\ \bar{w}_3 u_{20} &> \delta_2, \quad \delta_2 = \left(\|f_{3s}(k)\| + \left\|\tilde{\Delta}_3(k)\right\|\right) \\ f_s\left(\chi(k),k\right) &= -\chi^r(k)+\chi_d^r(k)+\tilde{f}_r\left(\chi(k),k\right)\end{aligned}$$

control (4.7) and (4.8) is able to drive the system (4.6) onto a small neighborhood of the sliding manifold $\varepsilon_2(k) = 0$ and $\varepsilon_3(k) = 0$. Hence, the armature and field control errors satisfies

$$\begin{aligned}|\varepsilon_2(k)| &= |\chi_2(k)-\chi_{2d}(k)| \leq O(T_s) \\ |\varepsilon_3(k)| &= |\chi_3(k)-\chi_{3d}(k)| \leq O(T_s).\end{aligned}$$

The sliding mode motion in a $O(T_s)$ boundary layer of $\varepsilon_2(k) = 0$ is described by the first order system:

$$\varepsilon_1(k+1) = \mathbf{k}_1\varepsilon_1(k)+\bar{w}_1\varepsilon_2(k)+\tilde{\Delta}_1(k) \tag{4.9}$$

By direct inspection of (4.9) or considering Proposition 4, there exists k_1 such that $\forall k > k_1$, the speed control error $\varepsilon_1(k)$ satisfies

$$\|\varepsilon_1(k)\| \leq \delta_1, \ \delta_1 = \frac{1}{\gamma}\left|\tilde{\Delta}_1(k)\right| + O(T_s).$$

where $O(T_s)$ is the boundary layer as defined in [46] and T_s is the sampling period.

It is worth to mention that the identification error $\tilde{\Delta}_1(k)$ can become arbitrarily small by adding more order terms into the neural identifier (4.1) and increasing the number of adjustable weights.

4.3 Electromagnetic Torque Control

Given full state measurements, the control objective is to develop the electromagnetic torque amplitude tracking of the DC motor (2.18) by the armature and field current amplitude tracking, as we see in the equation (2.16), using discrete-time SM control.

In (4.2) and (4.3), let us define $\chi_d^2(k) = [\chi_{2d}(k) \;\; \chi_{3d}(k)]^T$ as the armature and field current references, respectively. To control these variables, we define

$$\begin{aligned} \varepsilon^2(k) &= \chi^2(k) - \chi_d^2(k) \\ &= x^2(k) - \chi_d^2(k) + \Delta^2(k) \end{aligned} \tag{4.10}$$

where $\varepsilon^2(k) = [\varepsilon_2(k) \;\; \varepsilon_3(k)]^T$ $\chi^2(k) = [\chi_2(k) \;\; \chi_3(k)]^T$ $\Delta^2(k) = [\Delta_2(k) \;\; \Delta_3(k)]^T$; with $\varepsilon_2(k) = \chi_2(k) - \chi_{2d}(k)$, $\varepsilon_3(k) = \chi_3(k) - \chi_{3d}(k)$, and $\chi_d^2(k)$ is the desired value for $\chi^2(k)$. Applying the SM control technique, we have

$$\varepsilon^2(k+1) = \tilde{f}^2(\chi(k), k) + \tilde{w}^2 u(k) + \tilde{\Delta}^2(k) \tag{4.11}$$

where $\tilde{f}^2(\chi(k), k) = [\tilde{f}_2(\chi_2(k), k) \; \tilde{f}_3(\chi_3(k), k)]^T$ $u(k) = [u_1(k) \; u_2(k)]^T$, $\tilde{\Delta}^2(k) = [\tilde{\Delta}_2(k) \; \tilde{\Delta}_3(k)]^T$ $\tilde{f}_2(\chi_2(k), k) = \hat{w}_2(k)\hat{z}_2(\chi(k)) - \chi_{2d}(k+1)$, $\bar{w}^2 = [\bar{w}_2 \; \bar{w}_3]^T$ $\tilde{f}_3(\chi_3(k), k) = w_3(k) z_3(\chi_3(k)) - \chi_{3d}(k+1)$, and $\tilde{\Delta}^2(k) = [\Delta_2(k+1) \; \Delta_3(k+1)]^T$. Then, selecting the sliding variable as $s^2(k) = \varepsilon^2(k)$, where $s^2(k) = [s_2(k) \;\; s_3(k)]^T$ and taking into account the respective bounds $\|u_1\| \leq u_{01}$ and $\|u_2\| \leq u_{02}$ for the armature and field voltages respectively, the following control is proposed (2.67):

$$u(k) = \begin{cases} \tilde{u}_{eq}(k) & \text{for } \|\tilde{u}_{eq}(k)\| \leq u_0 \\ u_0 \frac{u_s}{\|u_s(k)\|} & \text{for } \|\tilde{u}_{eq}(k)\| > u_0 \end{cases} \tag{4.12}$$

where $\tilde{u}_{eq}(k) = [\tilde{u}_{1eq}(k) \; \tilde{u}_{2eq}(k)]^T$ $u_0 = [u_{01} \; u_{02}]^T$ $u_s = [u_{1s} \; u_{2s}]^T$ with

$$\begin{aligned} \tilde{u}_{eq}(k) &= -\bar{w}' \left[s^2(k) + \tilde{f}^2(\chi(k), k) \right] \\ &\quad + \bar{w}' \left[\tilde{f}^2(\chi(k-1), k-1) \right] + \tilde{u}_{eq}(k-1) \\ u_s(k) &= -\tilde{w}' \tilde{f}^2(\chi(k), k) \end{aligned}$$

where $\bar{w}' = [\bar{w}_2^{-1} \; \bar{w}_3^{-1}]^T$

Considering Proposition 3, under the condition

$$\bar{w}^2 u_0 > \delta, \quad \delta = \begin{bmatrix} \left(\|f_{2s}(k)\| + \left\| \tilde{\Delta}_2(k) \right\| \right) \\ \left(\|f_{3s}(k)\| + \left\| \tilde{\Delta}_3(k) \right\| \right) \end{bmatrix}$$
$$f_s(\chi(k), k) = -\chi^r(k) + \chi_d^r(k) + \tilde{f}_r(\chi(k), k)$$

control (4.12) is able to drive the system (4.11) onto a small neighborhood of the sliding manifold $\varepsilon^2(k) = 0$. Hence, the armature and field control errors satisfies

$$|\varepsilon_2(k)| = |\chi_2(k) - \chi_{2d}(k)| \leq O(T_s)$$
$$|\varepsilon_3(k)| = |\chi_3(k) - \chi_{3d}(k)| \leq O(T_s).$$

Considering Proposition 4, there exists k_1 such that $\forall k > k_1$, the electromagnetic torque control error $\varepsilon^2(k)$ satisfies

$$\left\| \varepsilon^2(k) \right\| \leq \delta, \quad \delta = \frac{1}{\gamma} \left| \tilde{\Delta}^2(k) \right| + O(T_s).$$

where $O(T_s)$ is the boundary layer as defined in [46] and T_s is the sampling period.

It is worth to mention that the identification error $\tilde{\Delta}^2(k)$ becomes arbitrarily small by adding more order terms into the neural identifier (4.1) and increasing the number of adjustable weights.

4.4 Simulations

For simulations, we consider a DC motor with separate winding excitation (2.19), whose parameters are listed in Table 4.1. These parameters correspond to a 5 Horse Power (HP) DC motor and are taken from [32].

4.4.1 Identification

Simulations are performed giving the same input for system (4.1) and for the plant model (2.19), using the parameters listed in Table 4.1. The results are included as follows: Fig. 4.1 presents angular speed identification, where $x_1(k)$ identifies $\chi_1(k) = \omega_m(k)$, with Fig. 4.2 showing the respective error $\chi_1(k) - x_1(k)$; Fig. 4.3 displays armature current identification, where $x_2(k)$ identifies $\chi_2(k) = i_a(k)$, with Fig. 4.4 portraiting the respective error $\chi_2(k) - x_2(k)$; and Fig. 4.5 shows the field current identification, where $x_3(k)$ identifies $\chi_3(k) = i_f(k)$, with Fig. 4.6 presenting the respective error $\chi_3(k) - x_3(k)$.

Parameter	Value	Description
R_a	1.6 Ω	Armature resistance
R_f	2500 Ω	Resistance of field winding
J_e	0.0315 $kg \cdot m^2$	Moment of inertia
L_a	16 mH	Armature inductance
L_f	156 mH	Field inductance
b_e	10^{-7} $\frac{kg \cdot m}{s}$	Viscous friction coefficient
L_{af}	1.976 H	Mutual inductance
T_L	7.81 $N \cdot m$	Nominal load
ω_n	183.25 $\frac{rad}{s}$	Nominal speed
u_a	200 V	Armature voltage
u_f	200 V	Field voltage
T_s	0.5 ms	Sampling period
P	5 HP	Electrical Power

Table 4.1: DC motor parameters.

Figs. 4.7, 4.8 and 4.9 show the evolution weights $w_{11}(k)$, $\hat{w}_2(k)$ and $w_{31}(k)$, respectively, for the identification process. Inputs u_a and u_f are defined as chirp functions. A chirp function is a sine wave whose frequency increases at a linear rate with time [32]. In this dissertation, a chirp function is used for excitation of the system dynamics.

As we can see in Figs. 4.1, 4.3, and 4.5, simulation results for identification are very encouraging; the response of the neural identifier (4.1) aproximates well the response of the plant model (2.19) in a short period of time.

4.4.2 Angular Speed Tracking

Tracking performance can be verified for the two plant outputs (angular speed and field current); simulations are performed for system (4.1), using the parameters listed in Table 4.1 and varying the load torque and the armature and field resistances, as shown in Figs. 4.10, 4.11, and 4.12, respectively. The controller is incepted at time equal to 0.5 seconds in order to let the system identification to converge. Simulations results are presented as follows: Fig. 4.13 shows the angular speed tracking performance, with $\omega_r(k)$ as the reference angular speed and $\omega_m(k)$ as the measured angular speed, Fig. 4.14 presents the respective error $\omega_r(k) - \omega_m(k)$; Fig. 4.15 displays the field current tracking performance with $i_{fr}(k)$ as the field current reference and $i_f(k)$ as the field current output, and with Fig. 4.16 showing the respective error $i_{fr}(k) - i_f(k)$; Fig. 4.10 displays the load torque

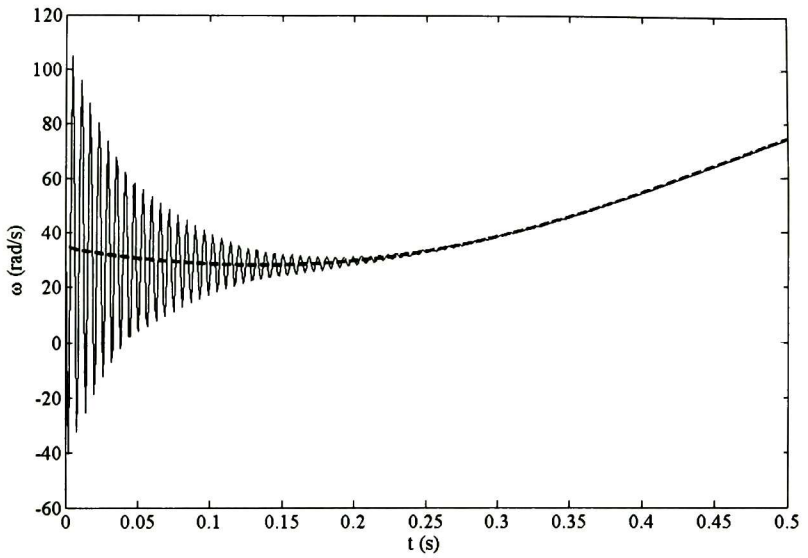

Figure 4.1: Angular speed identification: $x_1(k)$ (solid line), $\chi_1(k)$ (dashed line).

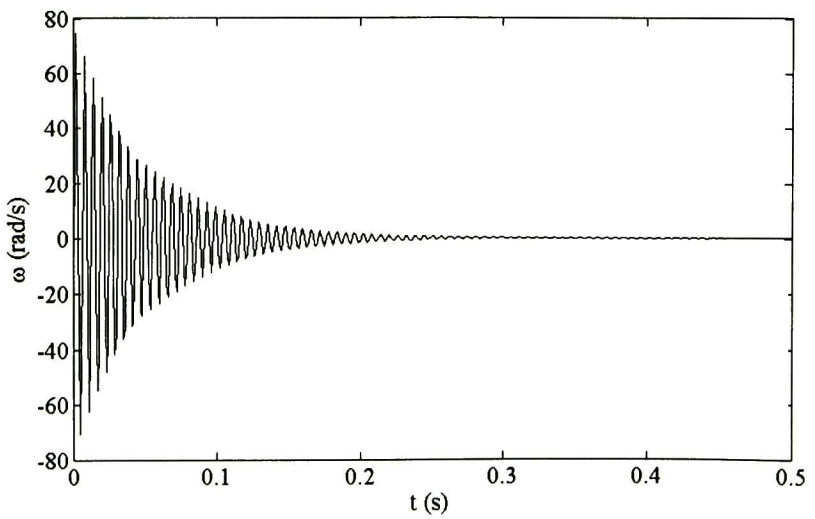

Figure 4.2: Angular speed identification error $\chi_1(k) - x_1(k)$.

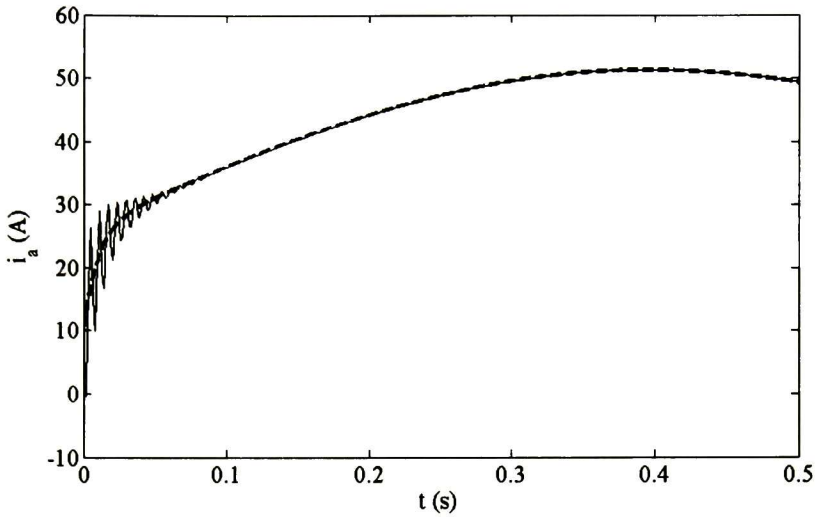

Figure 4.3: Armature current identification: $x_2(k)$ (solid line), $\chi_2(k)$ (dashed line).

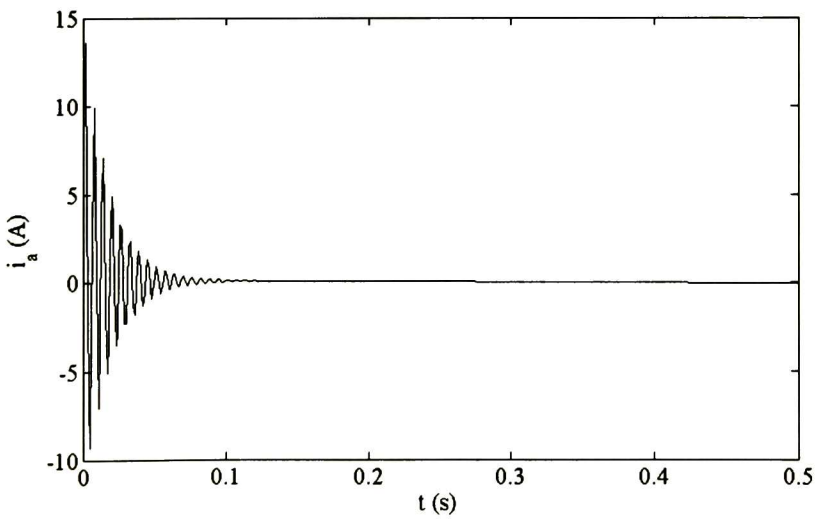

Figure 4.4: Armature current identification error $\chi_2(k) - x_2(k)$.

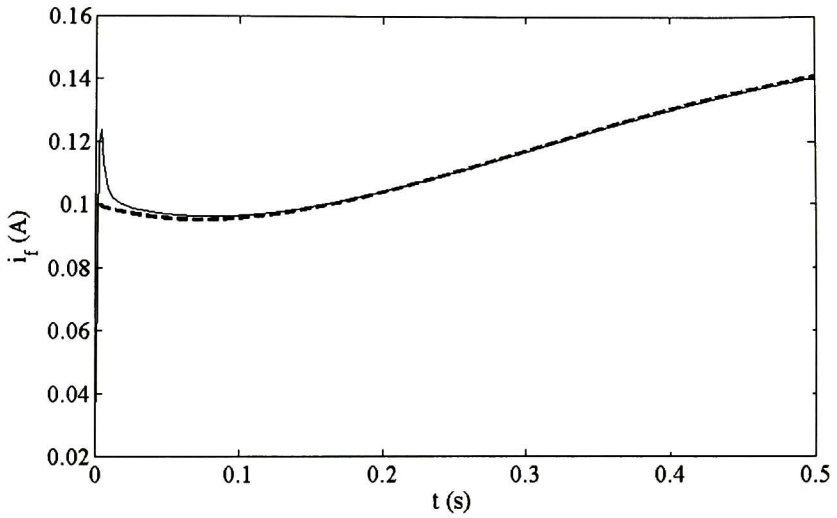

Figure 4.5: Field current identification: $x_3(k)$ (solid line), $\chi_3(k)$ (dashed line).

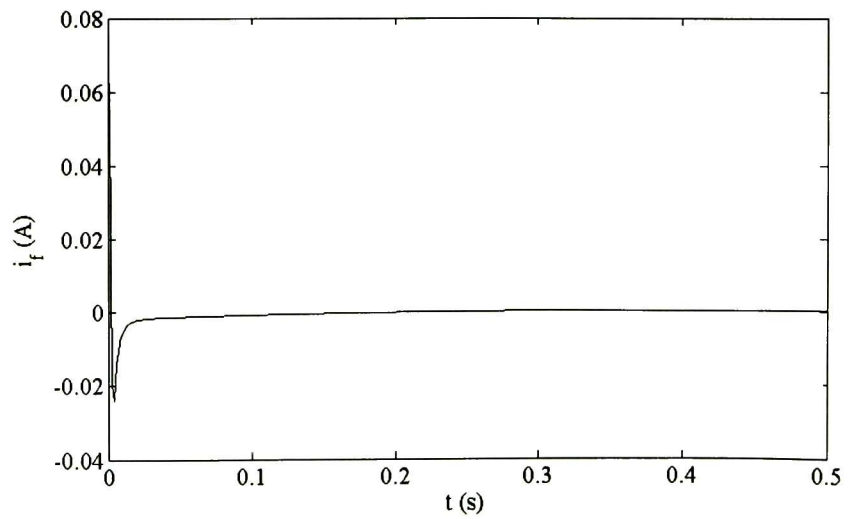

Figure 4.6: Field current identification error $\chi_3(k) - x_3(k)$.

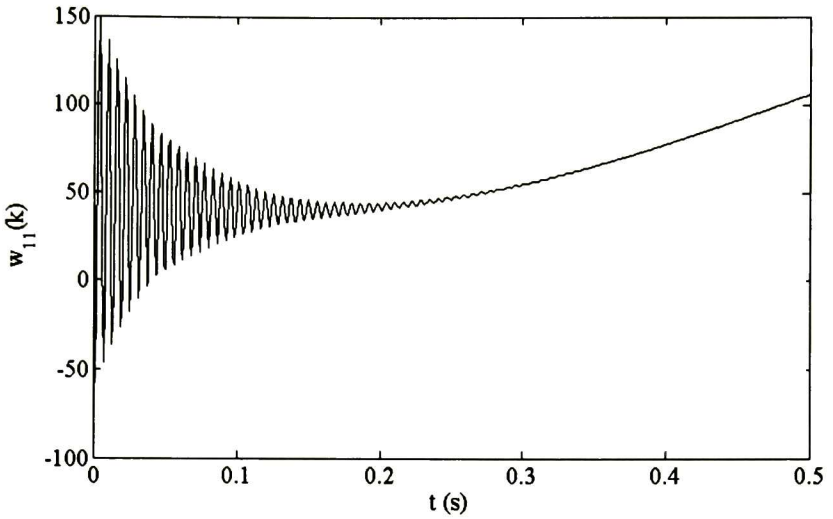

Figure 4.7: Evolution weights: $w_{11}(k)$

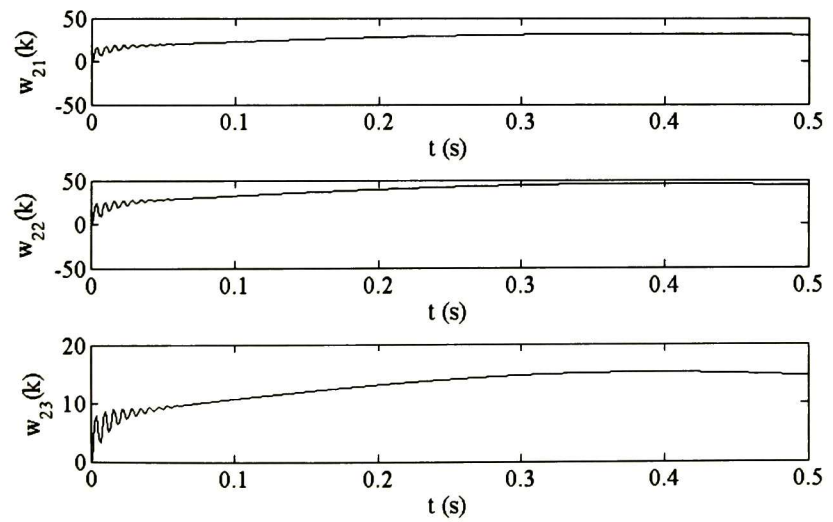

Figure 4.8: Evolution weights: $\hat{w}_2(k)$

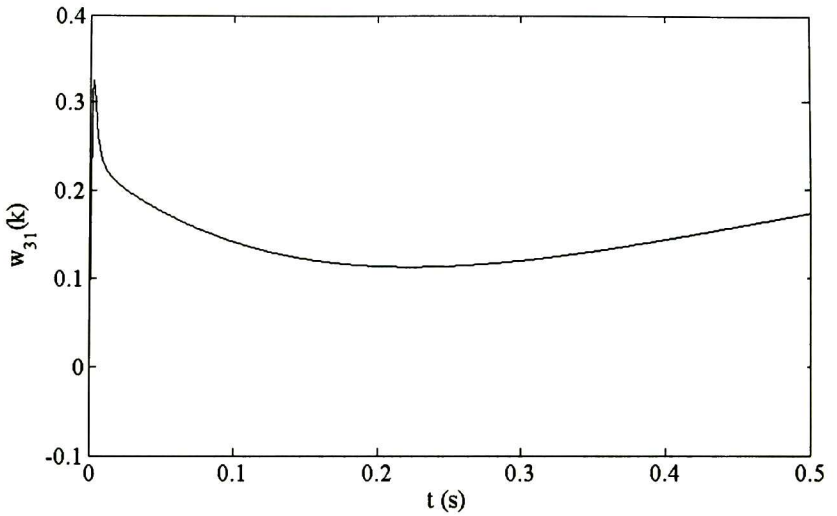

Figure 4.9: Evolution weights: $w_{31}(k)$

T_L applied as an external disturbance; Figs. 4.11 and 4.12 present parametric variation in the armature and field resistance (R_a) and (R_f) respectively; and Figs. 4.17, 4.18 and 4.19 show the evolution weights $w_{11}(k)$, $\hat{w}_2(k)$ and $w_{31}(k)$, respectively, for angular speed control process.

It is important to mention that in Figs. 4.13 and 4.15, we do not perceive oscillations as presented in the identification process; this is due to the fact that in these figures the identifier variables are not displayed.

Regarding Figs. 4.13 and 4.15, simulation results for angular speed and field current tracking present good behavior in presence of parameter variations and external disturbances. It is important to say that our scheme does not require to measure or to estimate the load torque.

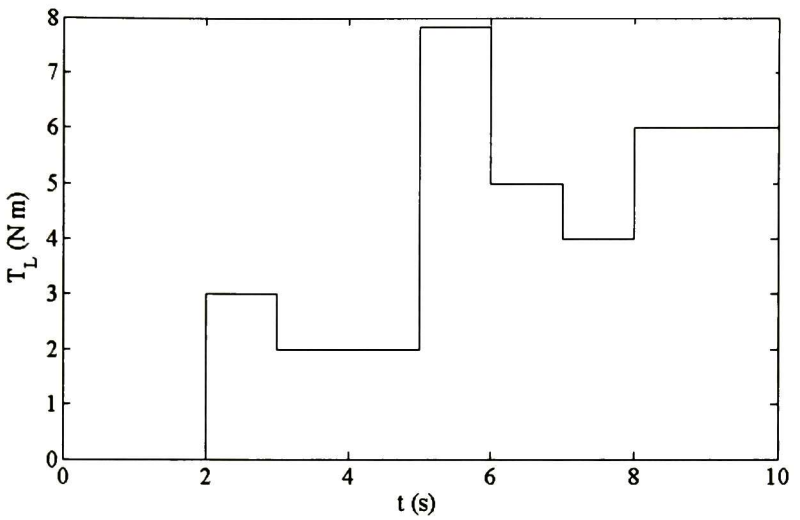

Figure 4.10: Load Torque T_L.

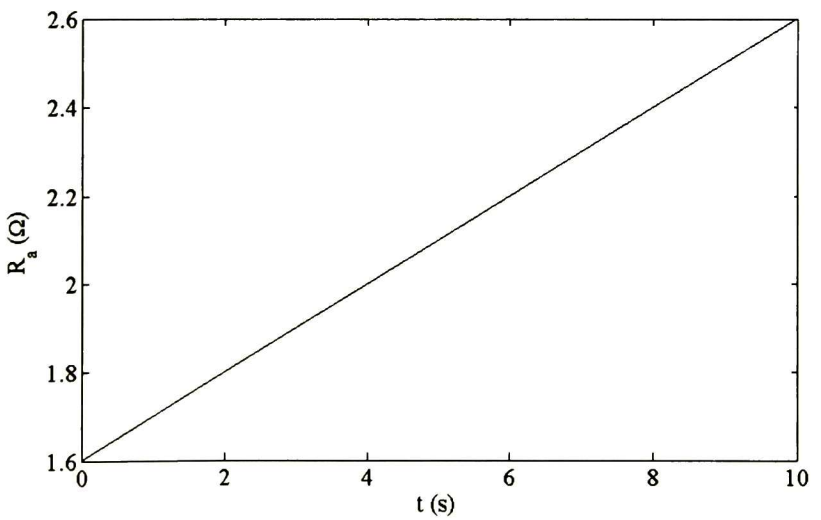

Figure 4.11: Armature resistance variation R_a.

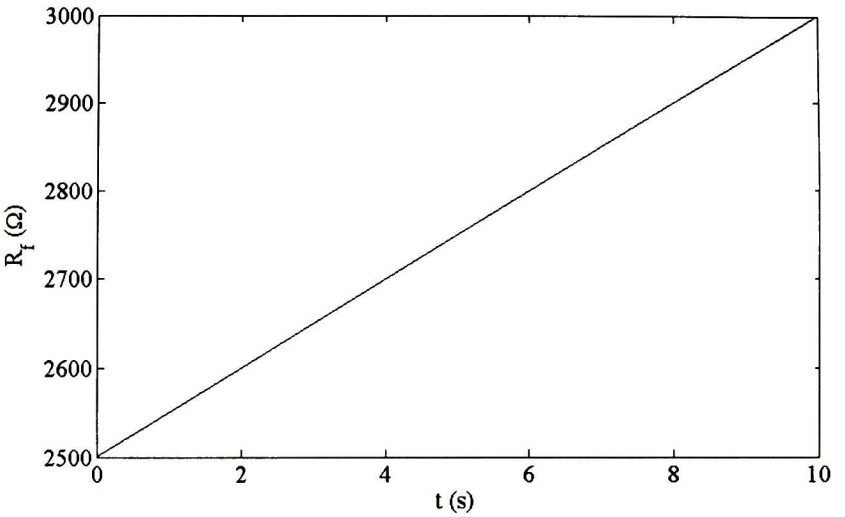

Figure 4.12: Field resistance variation R_f.

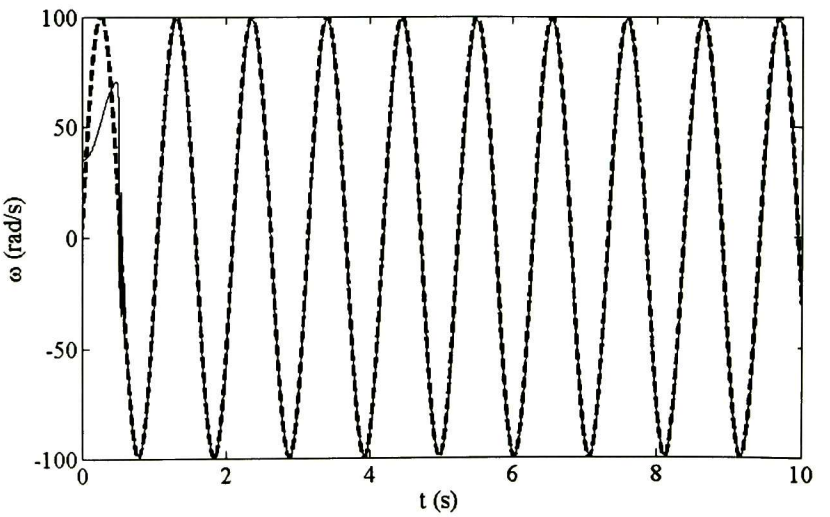

Figure 4.13: Angular speed tracking performance: $\omega_r(k)$ (dashed line) and $\omega_m(k)$ (solid line).

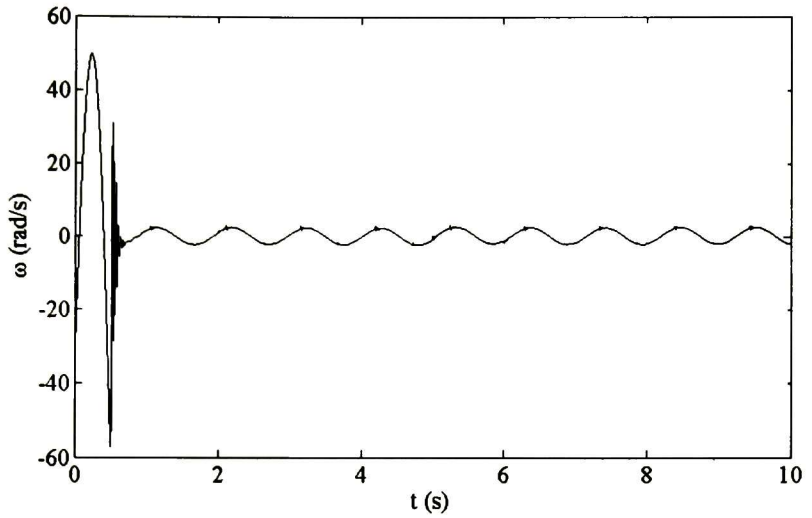

Figure 4.14: Angular speed tracking error $\omega_r(k) - \omega_m(k)$.

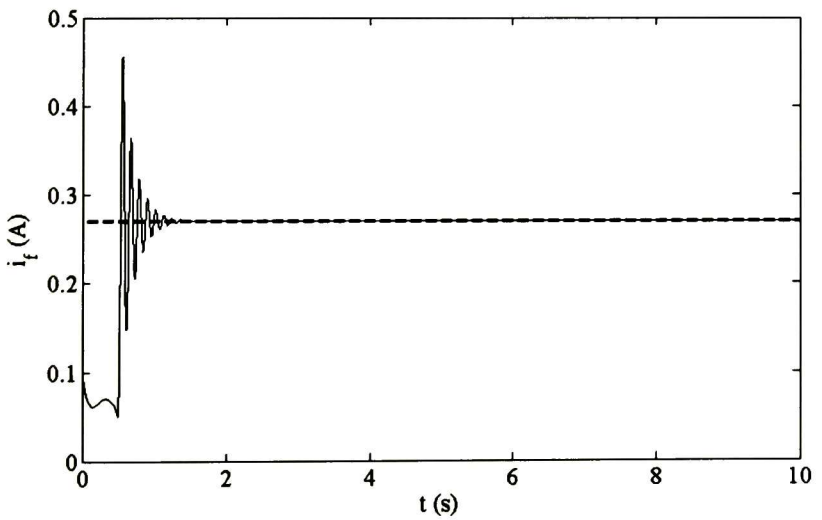

Figure 4.15: Field current tracking performance: $i_{fr}(k)$ (dashed line) and $i_f(k)$ (solid line).

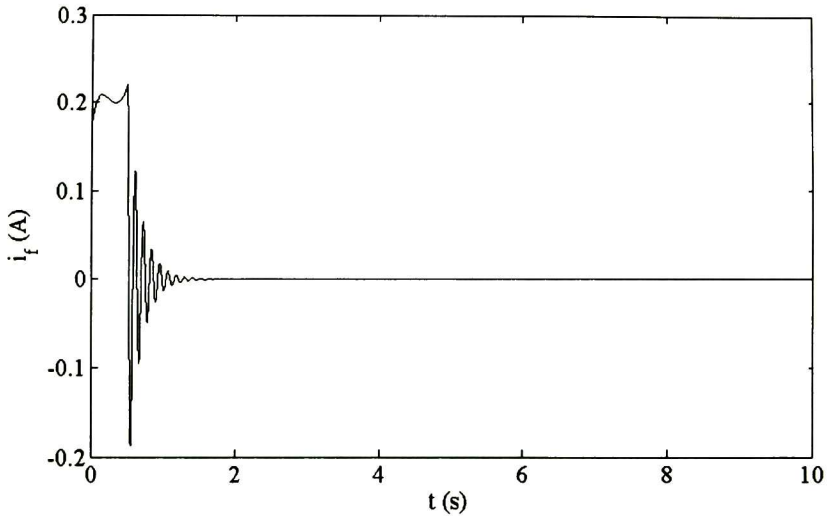

Figure 4.16: Field current tracking error $i_{fr}(k) - i_f(k)$.

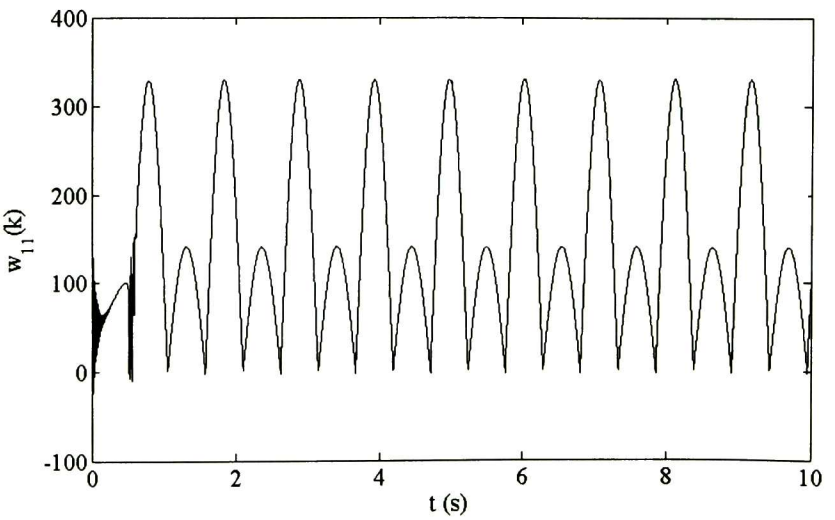

Figure 4.17: Evolution weights: $w_{11}(k)$

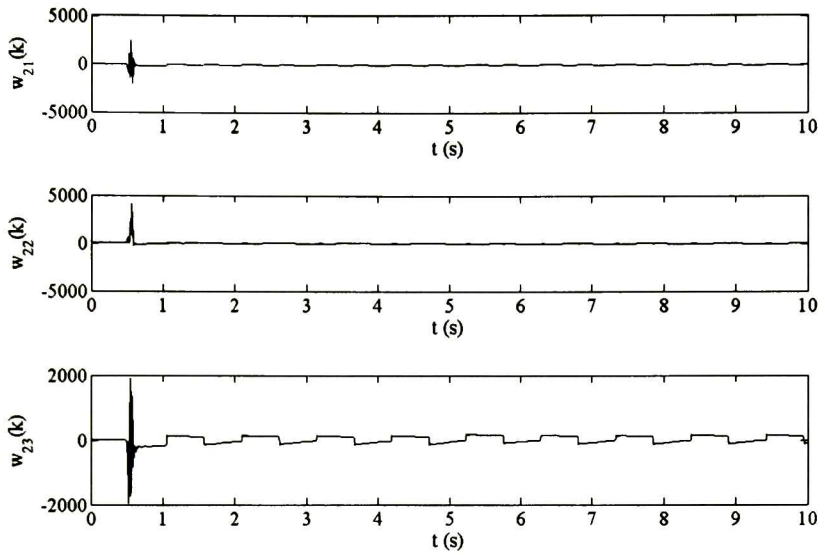

Figure 4.18: Evolution weights: $\hat{w}_2(k)$

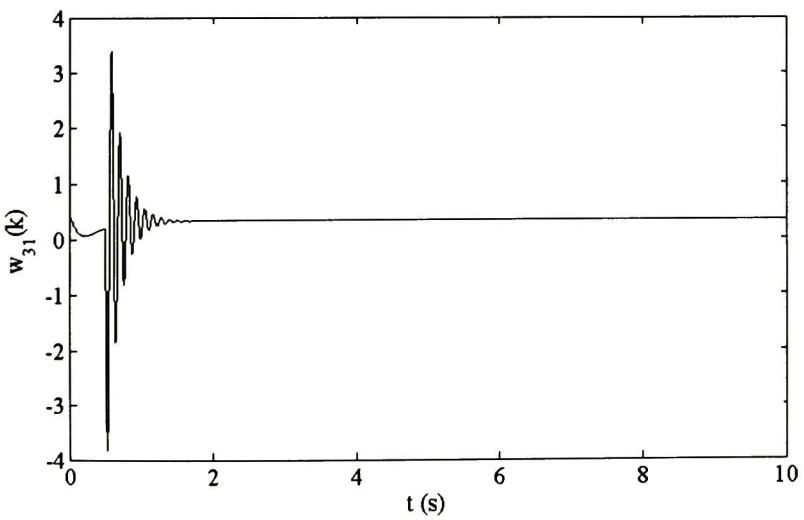

Figure 4.19: Evolution weights: $w_{31}(k)$

4.4.3 Electromagnetic Torque Tracking

Tracking performance is verified for the electromagnetic torque; the controller is incepted at time equal to 0.5 seconds in order to let the system identification to converge; the simulations results are presented as follows: Fig. 4.20 shows the electromagnetic torque tracking performance with $T_{er}(k)$ as the reference electromagnetic torque and $T_e(k)$ as the electromagnetic torque output and Fig. 4.21 displays the respective error $T_{er}(k) - T_e(k)$. Figs. 4.22, 4.23 and 4.24 show the evolution weights $w_{11}(k)$, $\hat{w}_2(k)$ and $w_{31}(k)$, respectively, for electromagnetic torque control process. These results are very encouraging.

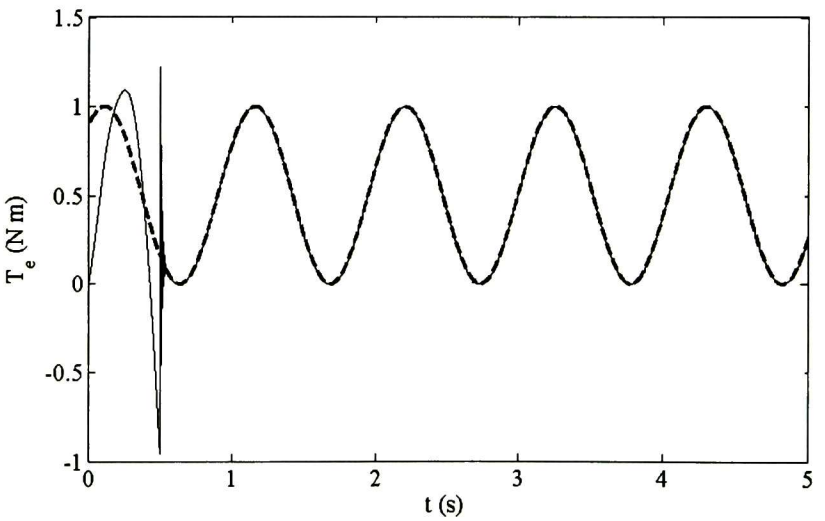

Figure 4.20: Electromagnetic torque tracking performance: $T_{er}(k)$ (dashed line) and $T_e(k)$ (solid line).

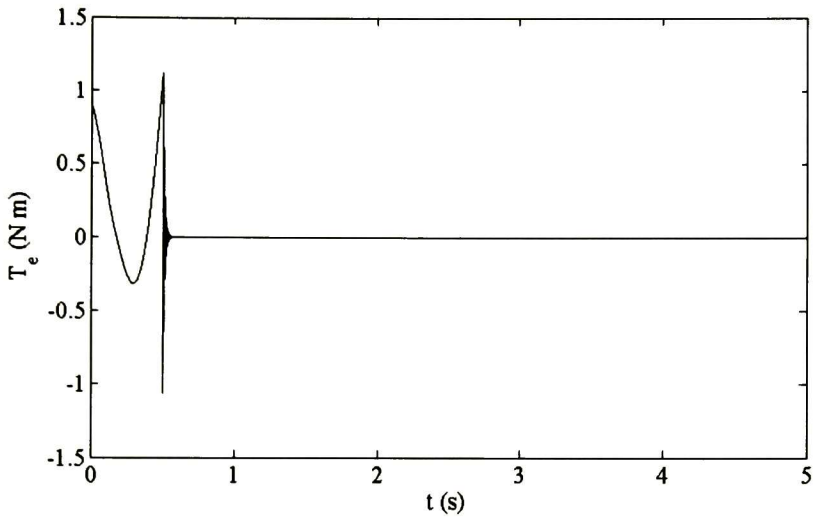

Figure 4.21: Electromagnetic torque tracking error $T_{er}(k) - T_e(k)$.

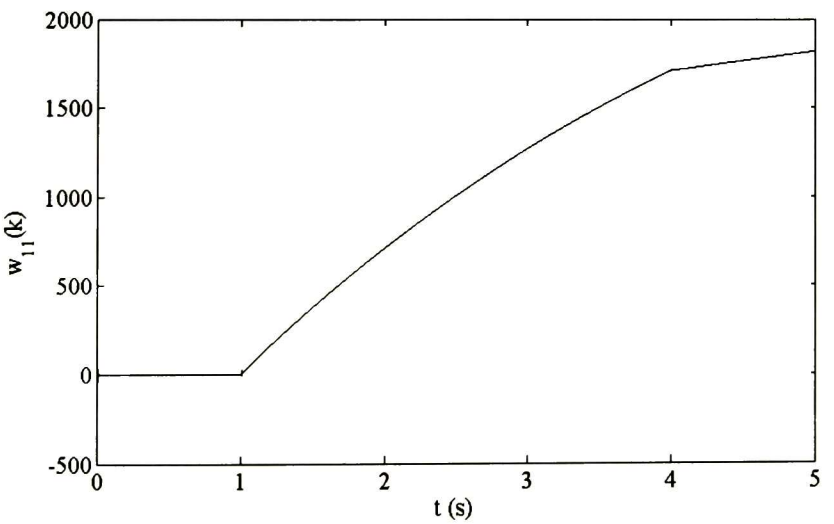

Figure 4.22: Evolution weights: $w_{11}(k)$

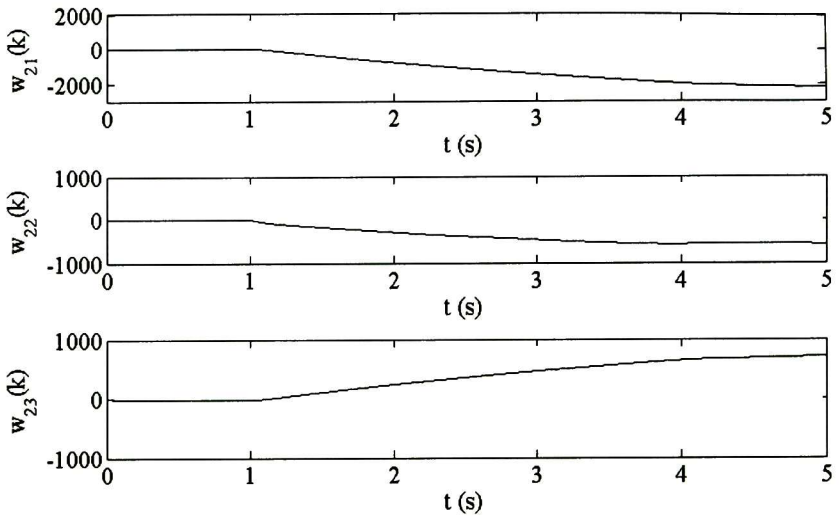

Figure 4.23: Evolution weights: $\hat{w}_2(k)$

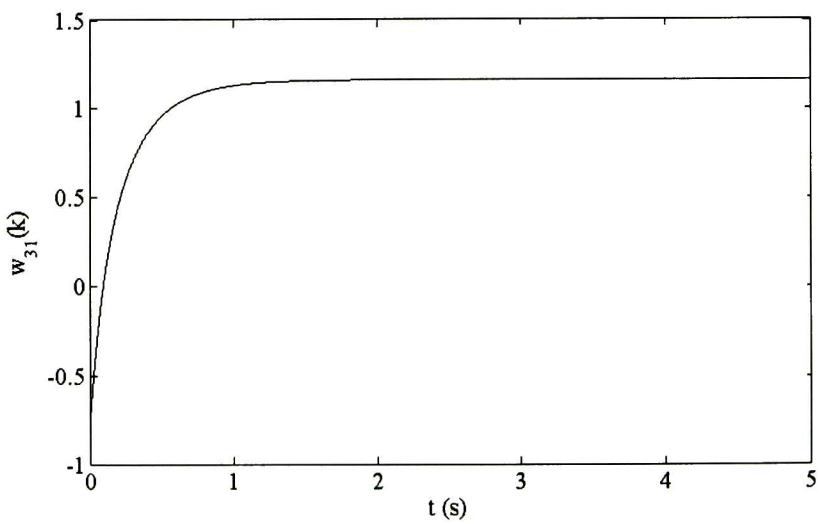

Figure 4.24: Evolution weights: $w_{31}(k)$

4.5 Real-Time results

Experimental tests are done using the prototype described in Chapter 3. Results presented in this section include: identification of the DC motor plant using the RHONN (4.1) identifier trained with the EKF (2.31 - 2.32); and angular speed and electromagnetic torque tracking in real-time. It is important to remark that for real-time implementation, we use a 0.25 HP DC motor, whose parameters are unknown. Experiments are performed with a sampling time of 0.5 ms.

4.5.1 Identification

During the identification process, the real plant and the NN (4.1) work in open-loop with the same input vector $\begin{bmatrix} u_a & u_f \end{bmatrix}^T$; u_a and u_f are defined as chirp functions with 90 and 100 V of amplitude, respectively, and a frequency from 1 Hz to 10 Hz for both signals. Identification results are included as follows: Fig. 4.25 presents the angular speed identification, with $x_1(k)$ that identifies $\chi_1(k) = \omega_m(k)$ and with Fig. 4.26 displaying the respective error $\chi_1(k) - x_1(k)$; Fig. 4.27 shows the armature current identification, with $x_2(k)$ that identifies $\chi_2(k) = i_a(k)$, and with Fig. 4.28 presenting the respective error $\chi_2(k) - x_2(k)$; and Fig. 4.29 shows the field current identification, with $x_3(k)$ that identifies $\chi_3(k) = i_f(k)$, with Fig. 4.30 displaying the respective error $\chi_3(k) - x_3(k)$. Figs. 4.31, 4.32 and 4.33 show the evolution weights $w_{11}(k)$, $\hat{w}_2(k)$ and $w_{31}(k)$, respectively, for identification process. The obtained results are very good.

4.5.2 Angular Speed Tracking

Tracking performance is verified for the two plant outputs $\omega_m(k)$ and $i_f(k)$. Real-time results are done with senoidal, triangular and a pulse train respectively, as trajectories to be tracked for the angular speed, as shown in Figs. 4.34, 4.40, and 4.46, where $\omega_r(k)$ is the reference speed and $\omega_m(k)$ is the speed output; additionally, their corresponding errors $\omega_r(k) - \omega_m(k)$ (Figs. 4.35, 4.41, and 4.47), as well as the control signals (Figs. 4.36, 4.42, and 4.48) for each case, are displayed. We incept disturbances at different times for each tracking case in order to verify the rapid drop recovery for the speed caused by these disturbances. The disturbances are created braking suddenly the motor shaft. Fig. 4.52 presents the field current tracking performance with $i_{fr}(k)$ as the reference field current and $i_f(k)$ as the field current output; Fig. 4.53 shows the field current tracking error $i_{fr}(k) - i_f(k)$. In Figs. 4.36, 4.42, 4.48 and 4.52 the controller is incepted at time equal to 3, 2, 3 and 2.5 seconds, respectively, in order to let the system identification to converge. The evolution weights $w_{11}(k)$, $\hat{w}_2(k)$ and $w_{31}(k)$ are shown in Figs. 4.37 - 4.39,

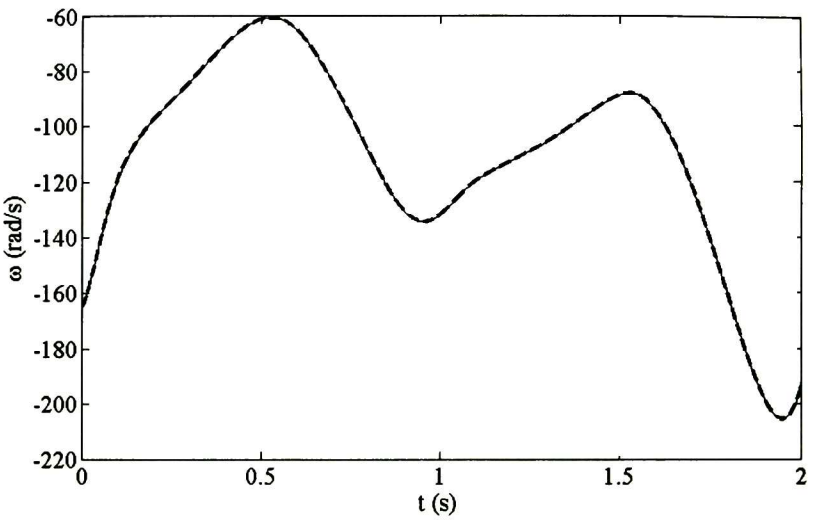

Figure 4.25: Angular speed identification: $x_1(k)$ (solid line), $\chi_1(k)$ (dashed line).

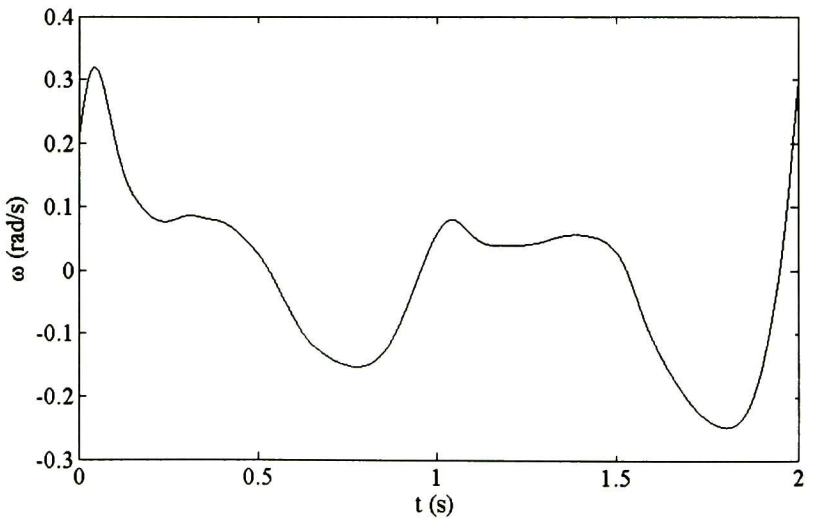

Figure 4.26: Angular speed identification error: $\chi_1(k) - x_1(k)$.

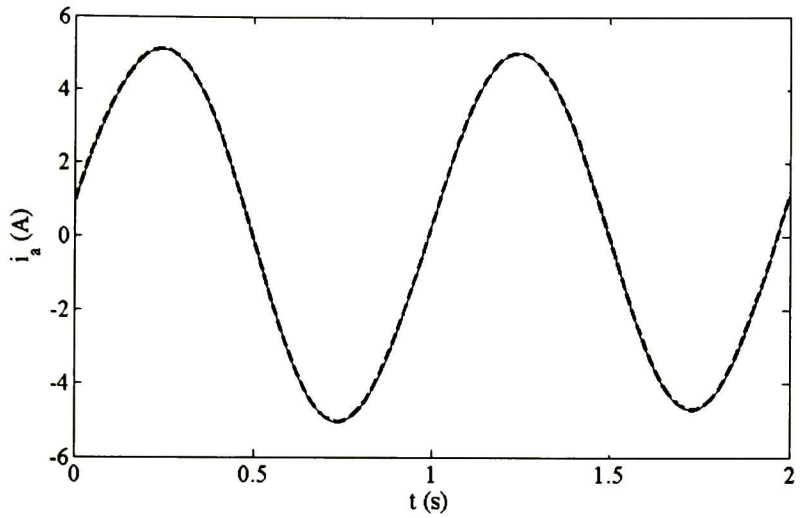

Figure 4.27: Armature current identification: $x_2(k)$ (solid line), $\chi_2(k)$ (dashed line).

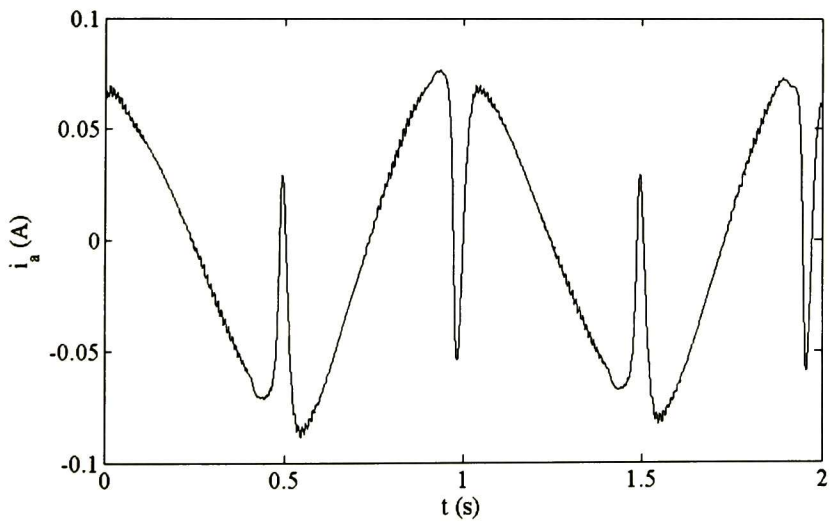

Figure 4.28: Armature current identification error: $\chi_2(k) - x_2(k)$.

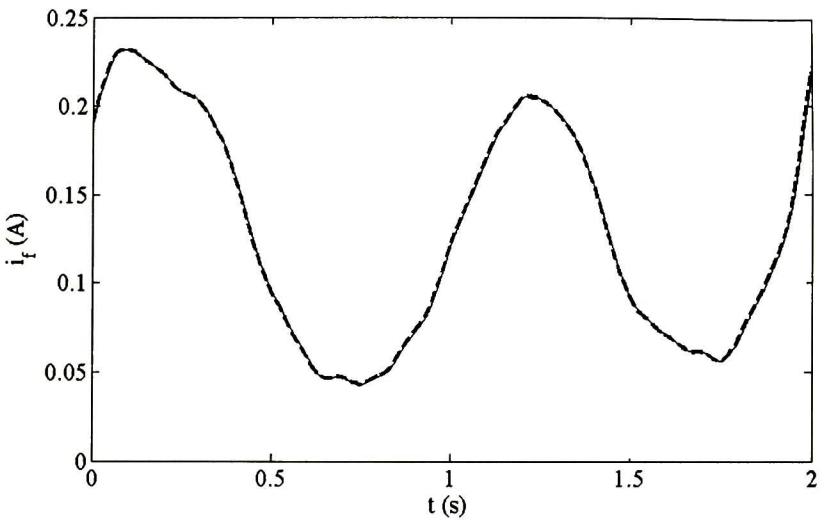

Figure 4.29: Field current identification: $x_3(k)$ (solid line), $\chi_3(k)$ (dashed line).

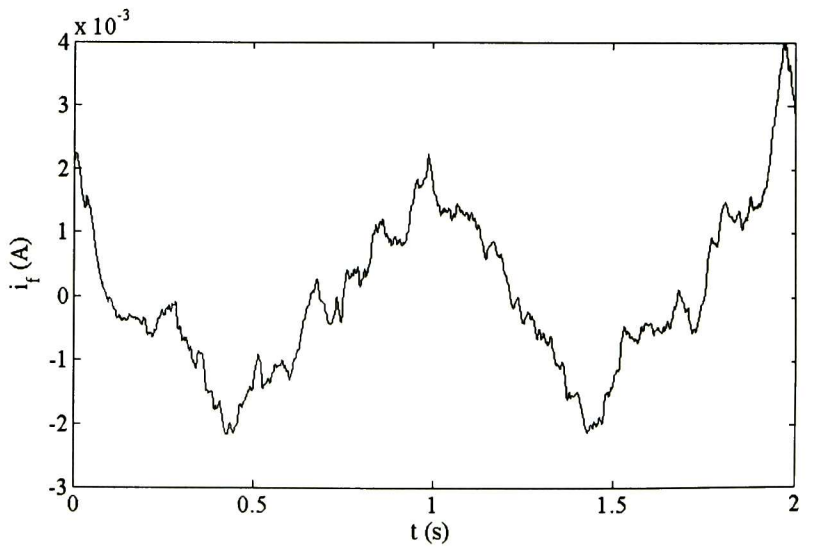

Figure 4.30: Field current identification error: $\chi_3(k) - x_3(k)$.

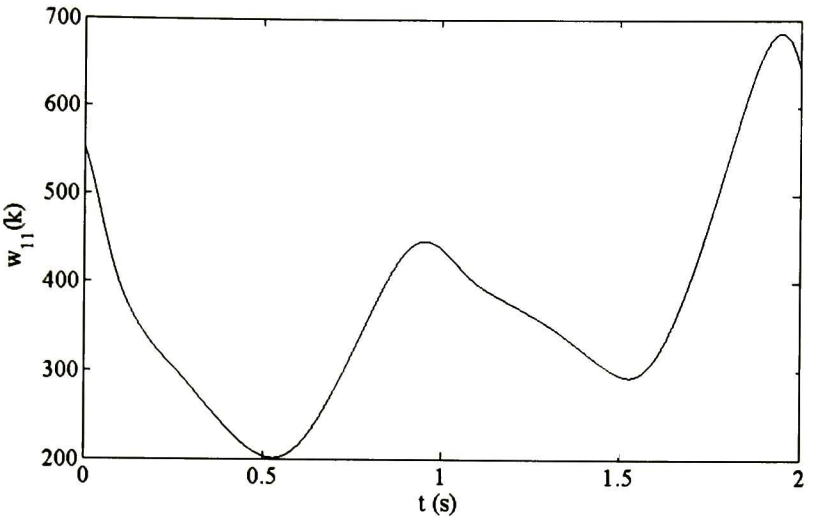

Figure 4.31: Evolution weights: $w_{11}(k)$

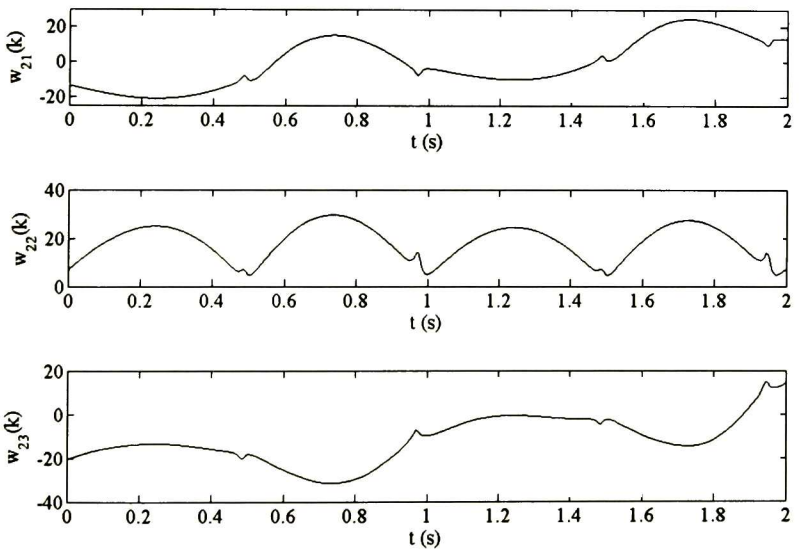

Figure 4.32: Evolution weights: $\hat{w}_2(k)$

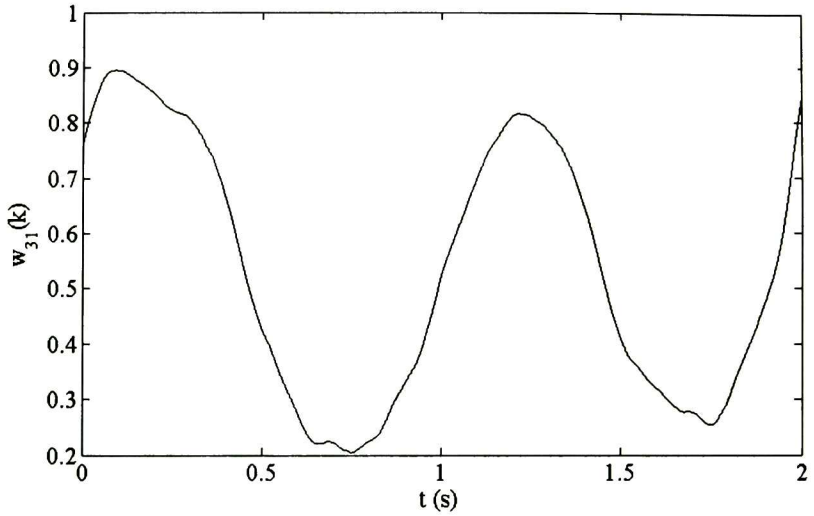

Figure 4.33: Evolution weights: $w_{31}(k)$

4.43 4.45 and 4.49 - 4.51 for each case of trayectory tracking (senoidal, triangular and a pulse train) respectively. The parameters of the neural network controller in the real time experiments are: $u_{01} = 90\ Volts$, $u_{02} = 100\ Volts$, $\mathbf{k}_1 = \mathbf{k}_2 = \mathbf{k}_3 = 0.9$, sampling time of 0.5 ms.

Regarding Figs. 4.34, 4.40, and 4.46, real-time results for angular speed tracking present good behavior and have a rapid recovery from speed drop caused by external load disturbances. It is worth to note that our scheme does not require to measure or to estimate the load torque and behave well even if the parameters are unknown; most of the existing publications can not handle these conditions.

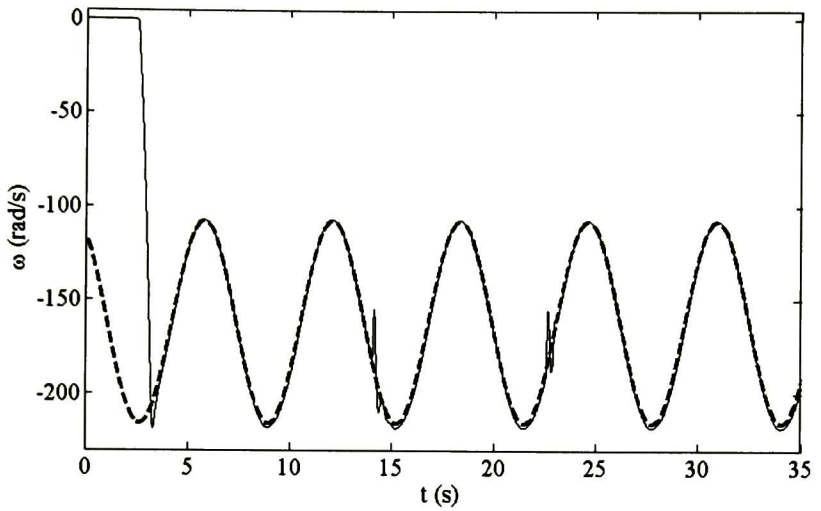

Figure 4.34: Speed tracking performance for a senoidal as the reference signal: $w_r(k)$ (dashed line), $w_m(k)$ (solid line).

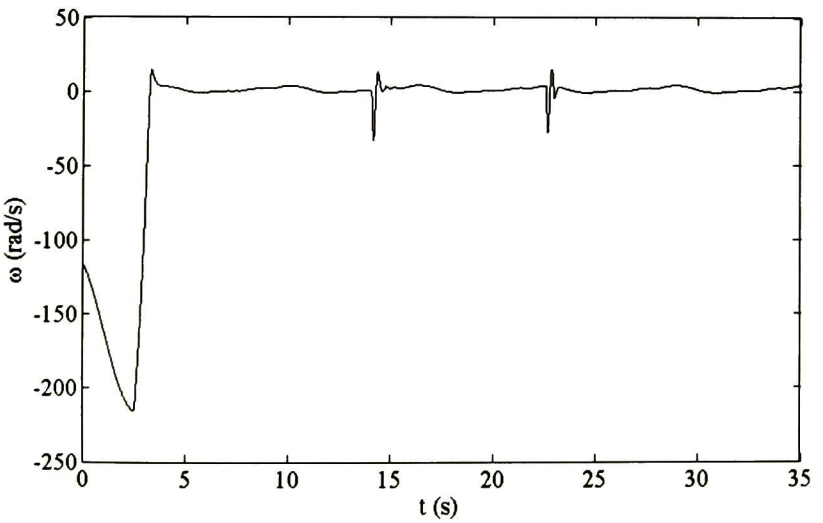

Figure 4.35: Angular speed tracking error for a senoidal as the reference signal: $\omega_r(k) - \omega_m(k)$.

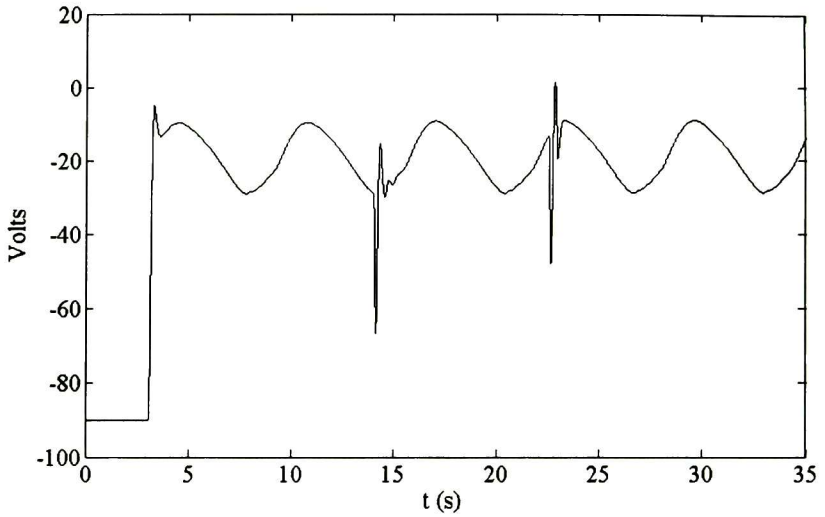

Figure 4.36: Control signal of the armature winding for a senoidal as the reference signal.

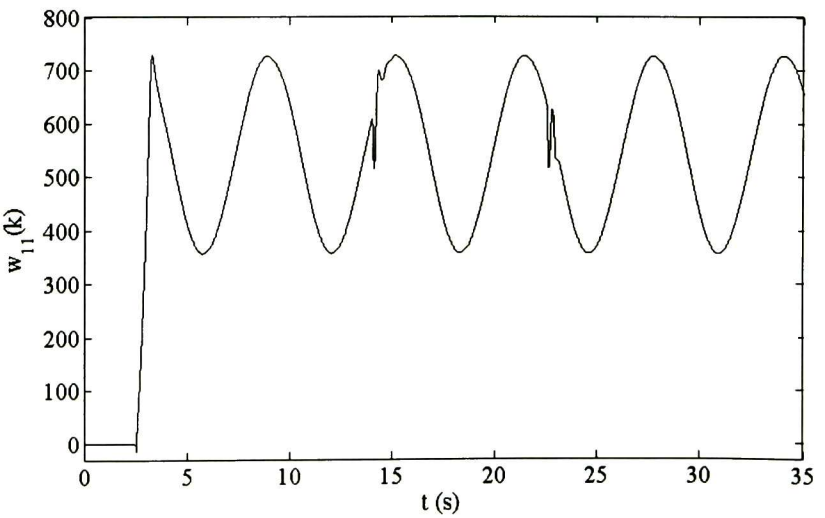

Figure 4.37: Evolution weights: $w_{11}(k)$

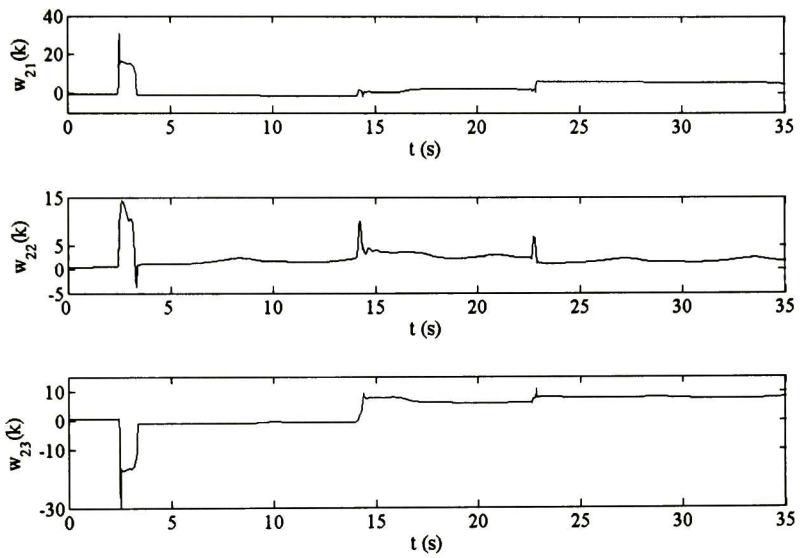

Figure 4.38: Evolution weights: $\hat{w}_2(k)$

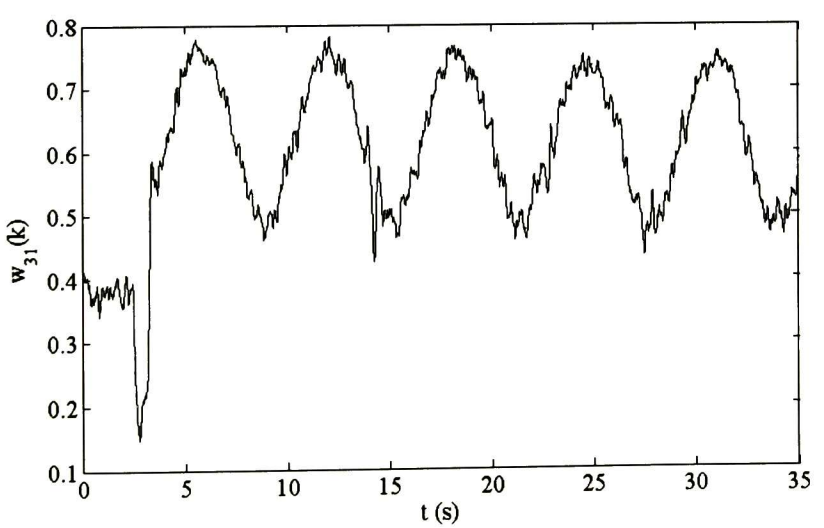

Figure 4.39: Evolution weights: $w_{31}(k)$

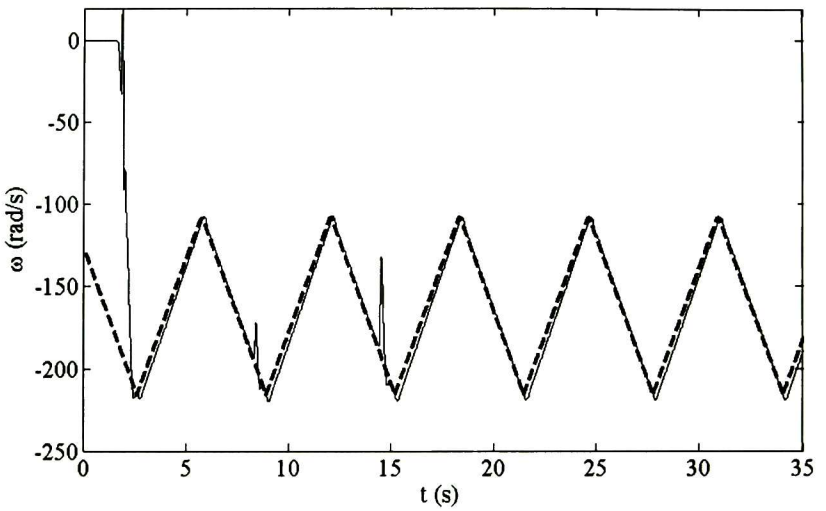

Figure 4.40: Speed tracking performance for a triangular as the reference signal: $w_r(k)$ (dashed line), $w_m(k)$ (solid line).

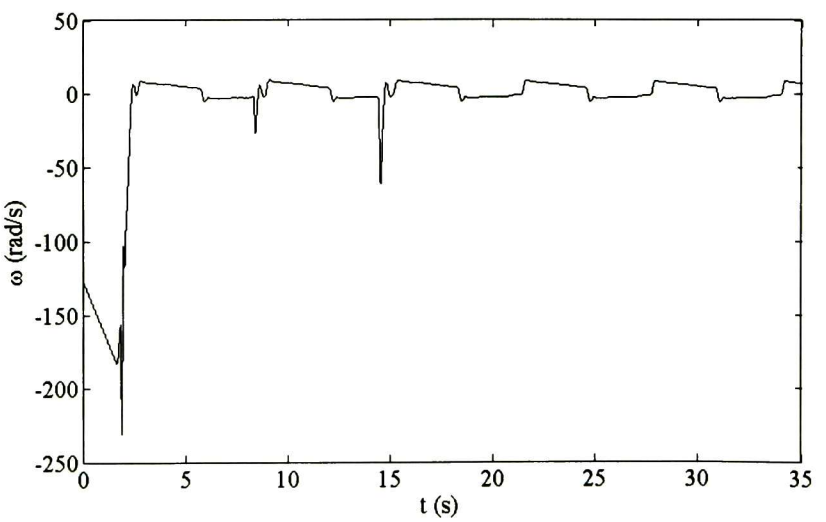

Figure 4.41: Angular speed tracking error for a triangular as the reference signal: $\omega_r(k) - \omega_m(k)$.

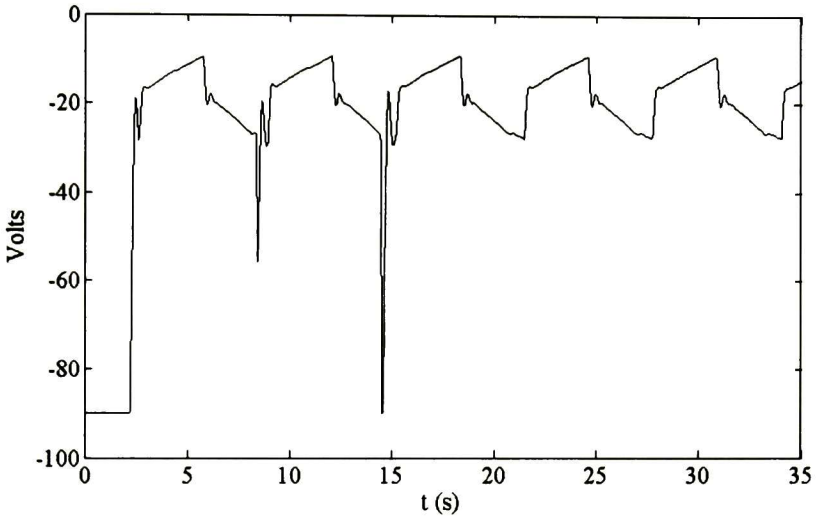

Figure 4.42: Control signal of the armature winding for a triangular as the reference signal.

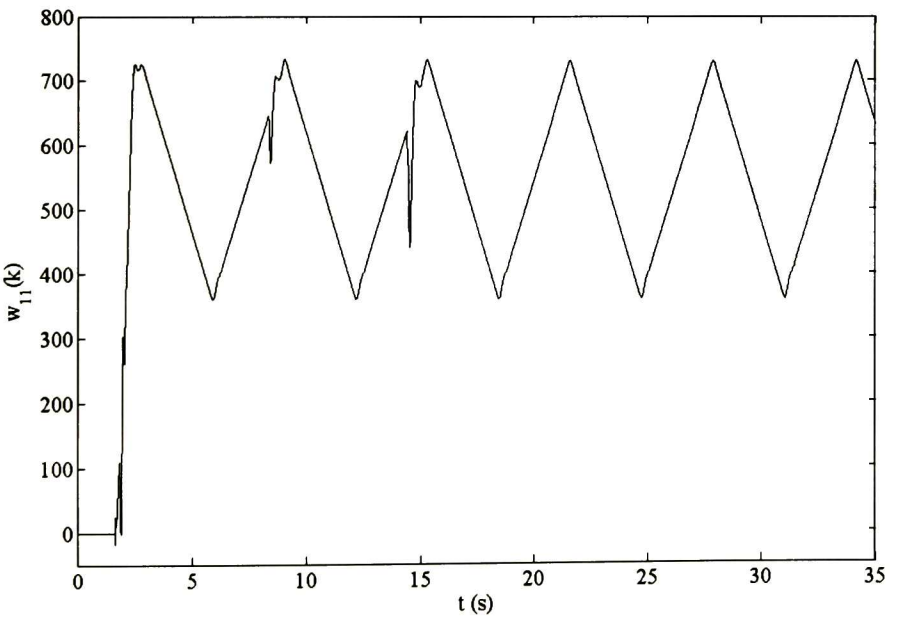

Figure 4.43: Evolution weights: $w_{11}(k)$

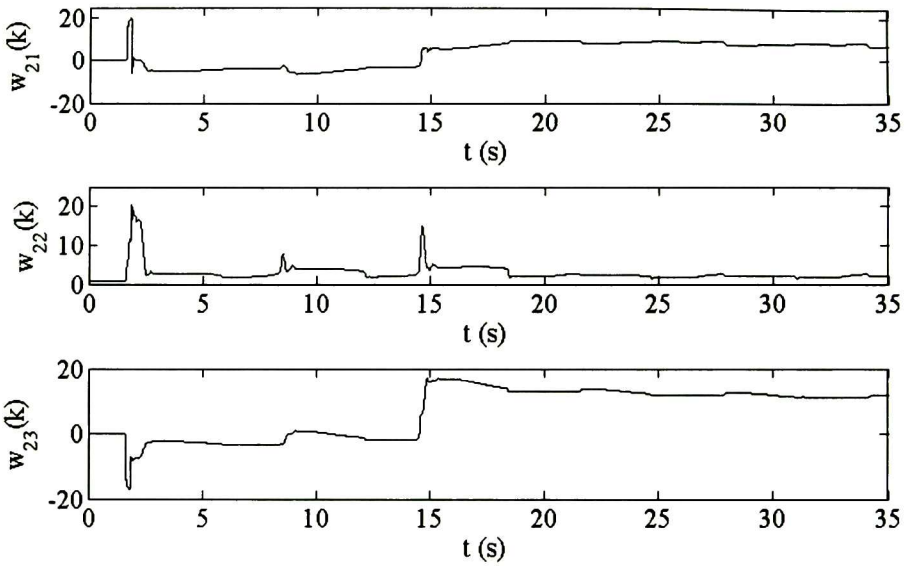

Figure 4.44: Evolution weights: $\hat{w}_2(k)$

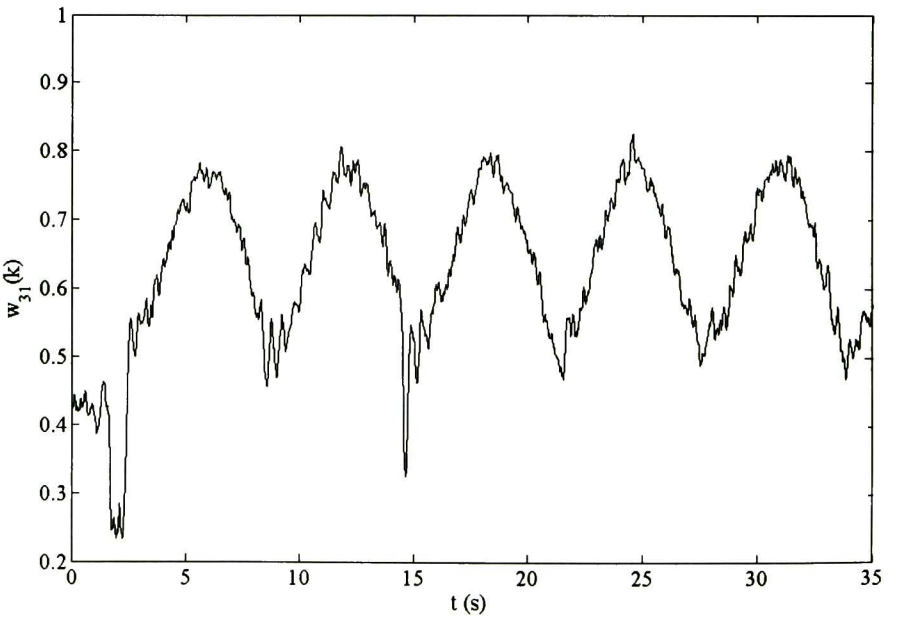

Figure 4.45: Evolution weights: $w_{31}(k)$

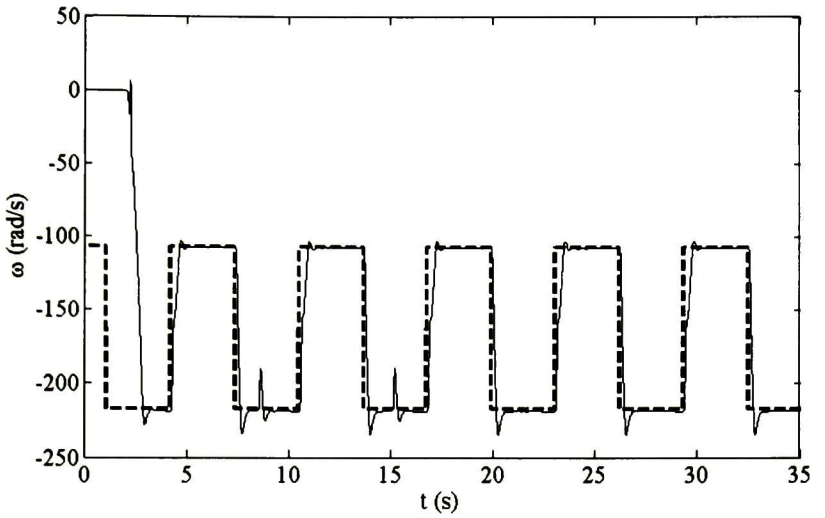

Figure 4.46: Speed tracking performance for a pulse train as the reference signal: $w_r(k)$ (dashed line), $w_m(k)$ (solid line).

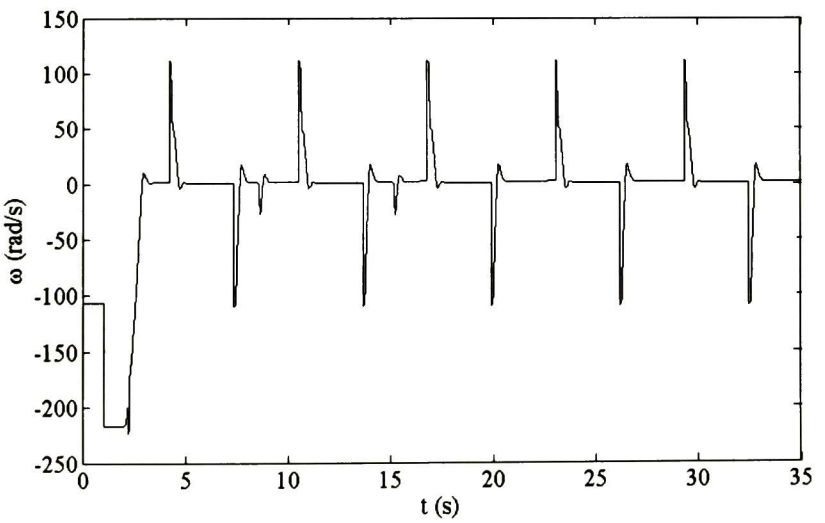

Figure 4.47: Angular speed tracking error for a pulse train as the reference signal: $\omega_r(k) - \omega_m(k)$.

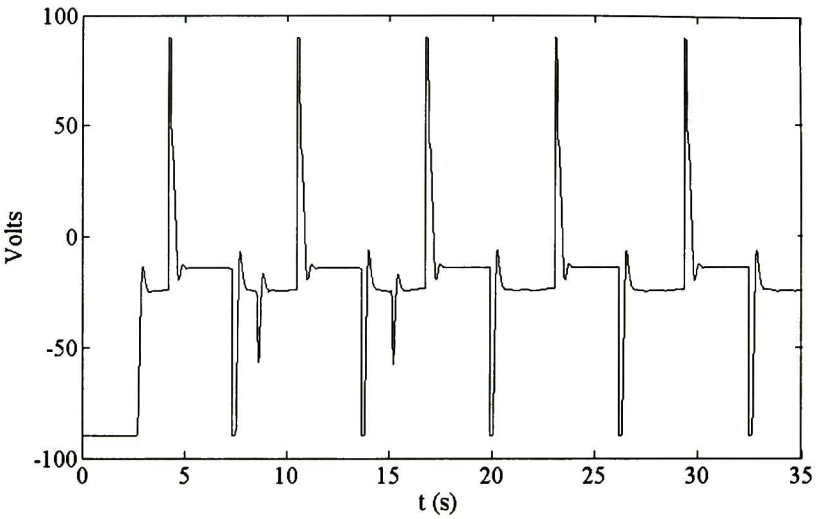

Figure 4.48: Control signal of the armature winding for a pulse train as the reference signal.

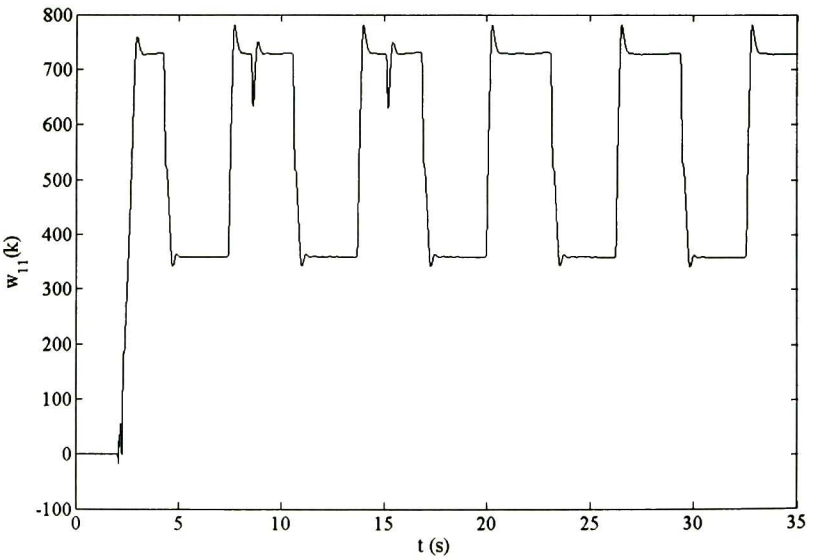

Figure 4.49: Evolution weights: $w_{11}(k)$

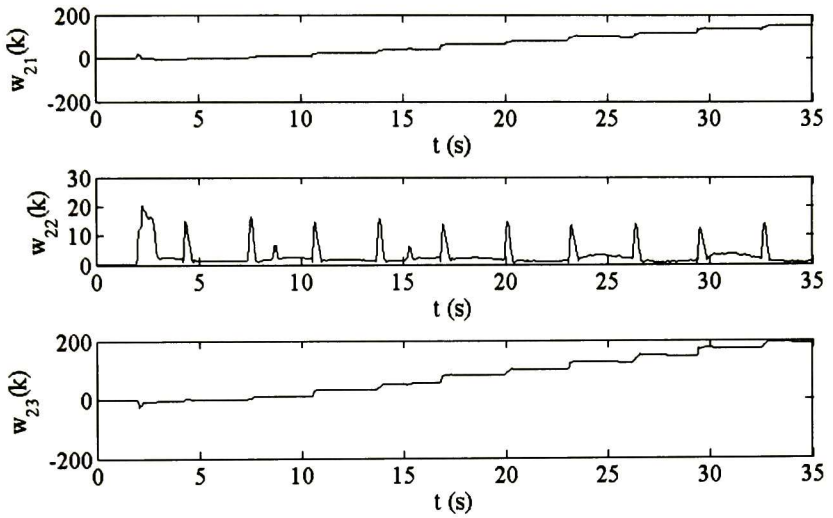

Figure 4.50: Evolution weights: $\hat{w}_2(k)$

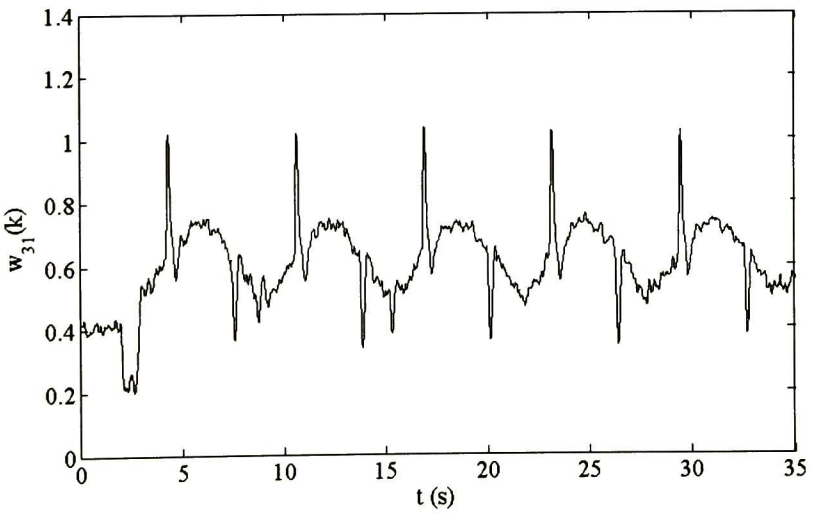

Figure 4.51: Evolution weights: $w_{31}(k)$

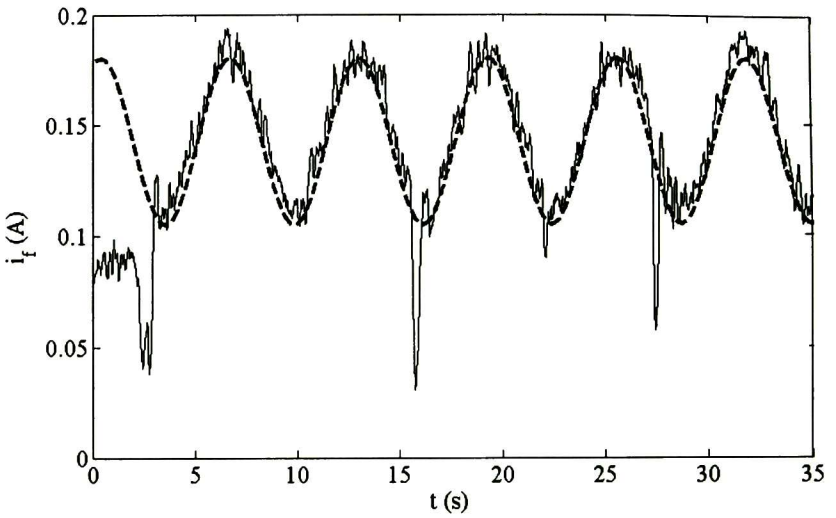

Figure 4.52: Field current tracking performance: $i_{fr}(k)$ (dashed line), $i_f(k)$ (solid line).

4.5.3 Electromagnetic Torque Tracking

Tracking performance is verified for the electromagnetic torque output controlling the two variables $i_a(k)$ and $i_f(k)$. The induction motor works at nominal angular speed; the electromagnetic torque controller of the DC motor is incepted at time equal to 1 second in order to let the system identification to converge. The electromagnetic torque controlled in the DC motor causes a reduction of the angular speed in the induction motor. Real-time results are presented as follows: Fig. 4.54 shows the electromagnetic torque tracking performance with $T_{er}(k)$ as the reference electromagnetic torque and $T_e(k)$ as the electromagnetic torque output; Fig. 4.55 presents the electromagnetic torque tracking error $T_{er}(k) - T_e(k)$; Fig. 4.56 displays the angular speed reduction in the induction motor caused by the electromagnetic torque controlled in the DC motor; and Figs. 4.57, 4.58 and 4.59 show the evolution weights $w_{11}(k)$, $\hat{w}_2(k)$ and $w_{31}(k)$, respectively, for elctromagnetic torque control process. The parameters of the neural network controller in the real time experiments are: $u_{01} = 90\ Volts$, $u_{02} = 100\ Volts$, $\mathbf{k}_1 = \mathbf{k}_2 = \mathbf{k}_3 = 0.9$, sampling time of $0.5\ ms$.

Note that T_e depends on the value of L_{af}. In this case, it can be observed that if the exact value of L_{af} is not known, the problem of tracking the electromagnetic torque reference, becomes a problem of tracking an uncertain signal. This is a very difficult problem in general. Some works recently appeared deal with this problem in the case

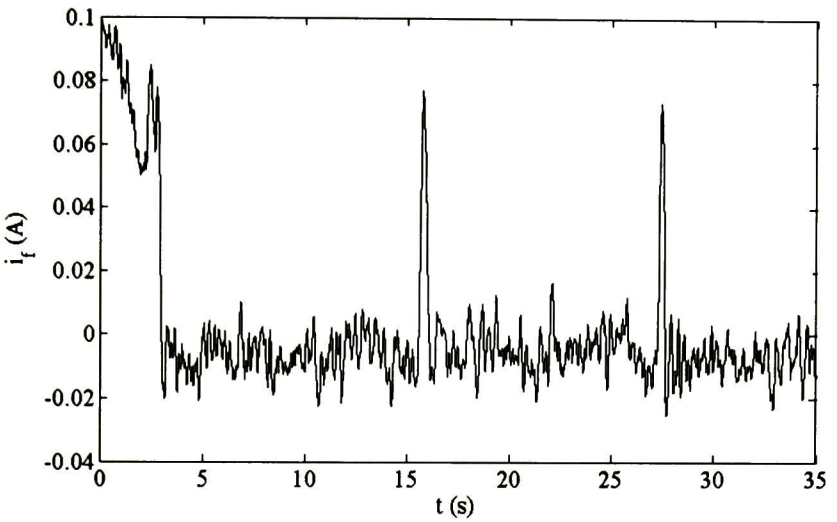

Figure 4.53: Field current tracking error: $i_{fr}(k) - i_f(k)$.

when the reference signal is generated by uncertain linear system, the so called exosystem. This is known as the robust regulation problem with uncertain exosystems ([6], [30], [31], [33], [40]); however, to the best of our knowledge, the general case is still an open problem.

Regarding to Fig. 4.54, real-time results for electromagnetic torque tracking are very good. It is worth to note that our scheme behaves well even if the parameters are unknown.

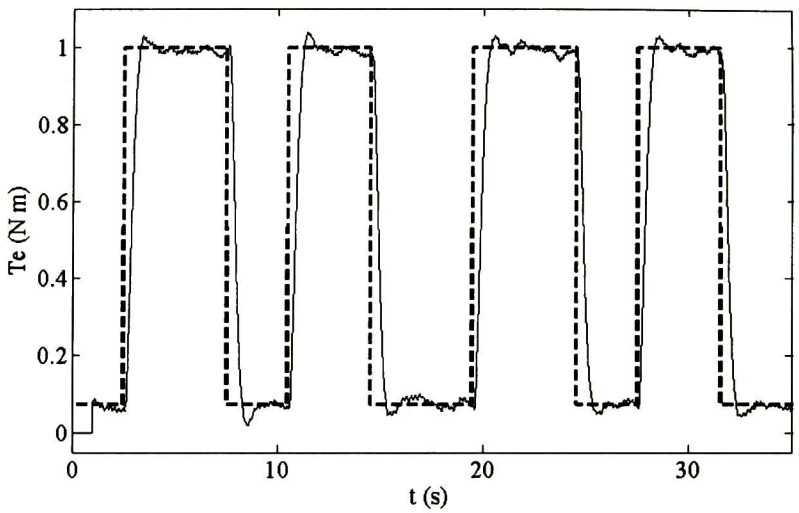

Figure 4.54: Electromagnetic Torque tracking: $T_{er}(k)$ (Electromagnetic torque reference in dashed line), $T_e(k)$ (Electromagnetic torque output in solid line).

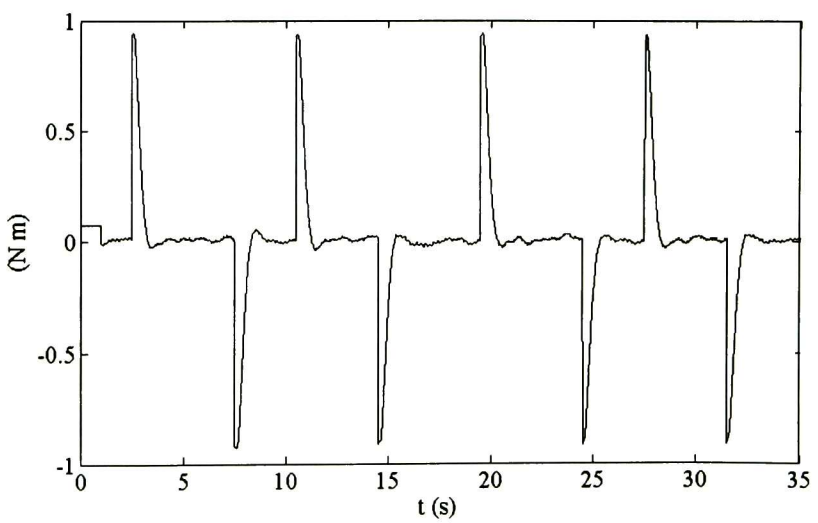

Figure 4.55: Electromagnetic torque tracking error $T_{er}(k) - T_e(k)$.

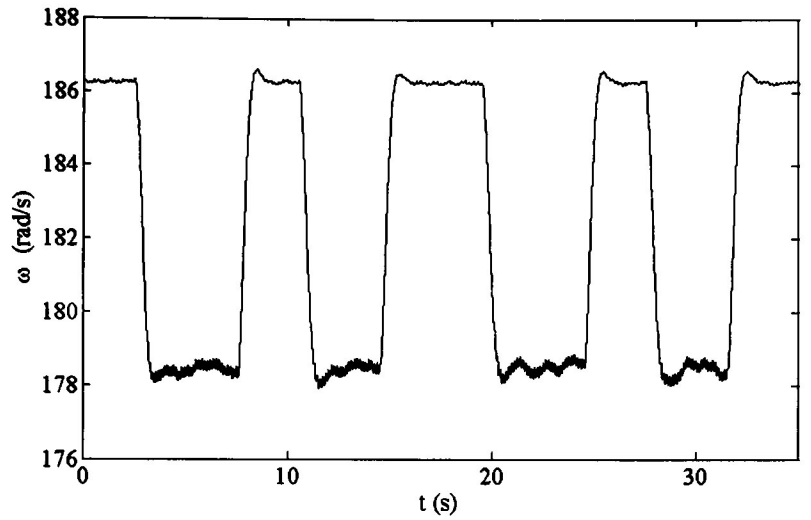

Figure 4.56: Angular speed response of the induction motor.

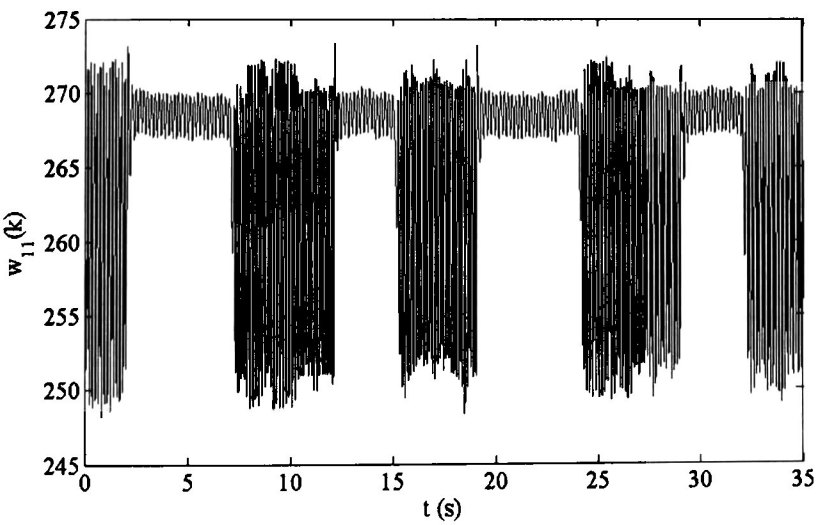

Figure 4.57: Evolution weights: $w_{11}(k)$

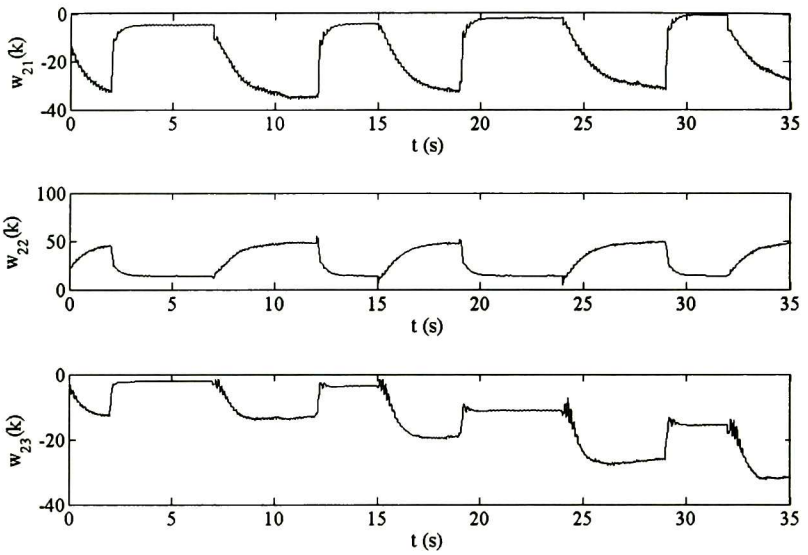

Figure 4.58: Evolution weights: $\hat{w}_2(k)$

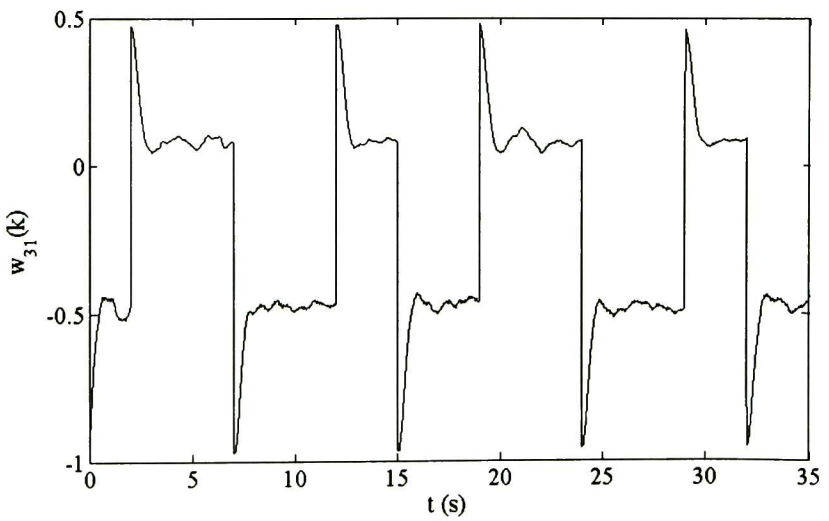

Figure 4.59: Evolution weights: $w_{31}(k)$

Chapter 5

Conclusions and Future Work

5.1 Conclusions

In the present dissertation, a method to design a robust indirect control for a class of MIMO discrete-time nonlinear (particularly to a DC motor with separate winding excitation) uncertain system is developed. This method proposes a control scheme which can be applied with full state measurement; additionally, it includes:

1. A discrete-time recurrent high order neural network in order to identify the plant model is proposed. The proposed neural identifier is in the nonlinear block controllable form, which allows to apply the block feedback linearization control and sliding modes techniques. This neural identifier is trained with the EKF algorithm using a series-parallel configuration.

2. Based on the neural identifier model and applying the discrete-time block control approach, a nonlinear sliding manifold with a desired asymptotically stable motions, is formulated. Using a Lyapunov functions approach, a discrete-time sliding mode control which makes the designed sliding manifold attractive, is introduced.

3. A prototype is integrated in order to test in real-time the proposed identifier and control scheme.

4. Effectiveness of this scheme are illustrated via computer simulations and real-time implementation for two cases: the angular speed and field current controller, and the electromagnetic torque controller.

The combine approach, which includes identification of the plant with the neural network and application of the block control and sliding modes control techniques, ensure

robustness with respect to external disturbances and parameter variation and stable desired sliding motion over the designed proposed sliding manifold.

The advantages of this control scheme with respect to other controllers (particularly controlling the angular speed and the electromagnetic torque of the DC motor with separate winding excitation) are: this scheme does not require the knowledge of the plant parameters, neither the load torque; this scheme is developed and analyzed in discrete-time; also the validation of these analysis is shown with real-time results. Furthermore, we develop a control scheme for electromagnetic torque even that there exist no many publications for controlling the electromagnetic torque; the only available is [17], to the best of our knowledge. Additionally, this control technique can be applied to systems that are or can be transformed to the NBC form which is possible to apply the procedure presented in this dissertation. And finally, in this control scheme and in the stability analysis, a method for reducing the effect of unknown terms is considered.

5.2 Future work

Researches will be continue along the following lines:

1. To develop a similar scheme for angular position.
2. To develop robust direct neural controller based on the backstepping technique, approximated by a high order neural network, for angular speed, electromagnetic torque and angular position.
3. To perform real-time implementation of these new controllers.
4. To compare the results of the neural networks control schemes with the block control and backstepping control schemes.

References

[1] K. Abidi and A. Šabanovic, "Sliding-mode control for high-precision motion of a piezostage", *IEEE Transactions on Industrial Electronics*, vol. 54, no. 1, pp. 629–637, 2007.

[2] A. Y. Alanis, E. N. Sanchez, and A. G. Loukianov, "Discrete-time recurrent neural induction motor control using Kalman learning" *Proceedings of the 2006 International Joint Conference on Neural Networks*, Vancouver, Canada, 2006.

[3] Baldor Electric Company, "Product Information Packet CD3425" Baldor Reliance (http://www.baldor.com), Fort Smith, AR., USA, 2007.

[4] BEI Industrial Encoders, "BEI Model H25 Incremental Encoder" BEI Design and Specifying Guide (http://www.beiied.com), Goleta, CA., USA, 2007.

[5] F. Betin, D. Pinchon, and G. A. Capolino, "A time-varying sliding surface for robust position control of a DC motor drive" *IEEE Transactions on Industrial Electronics*, vol. 49, no. 2, pp. 462–473, 2002.

[6] M. Bodson and S. Douglas, "Adaptive algorithms for the rejection of sinusoidal disturbances with unknown frequency" *Automatica*, vol. 33, pp. 2213–2221, 1997.

[7] C. K. Chui and G. Chen, *Kalman Filtering with Real-Time Applications*, 3rd ed., Springer-Verlag, New York, N.Y., USA, 1998.

[8] B. Drazenovic, "The invariance conditions for variable structure systems", *Automatica*, vol. 5, pp. 287–295, 1969.

[9] DsPace, "DS1104 R&D Controller Board" Catalog dSPACE DS1104 Installation and Configuration (http://www.dspace.com), Padernborn, N.W., Germany, 2006.

[10] R. A. Felix, E. N. Sanchez, and A. G. Loukianov, "Avoiding controller singularities in adaptive recurrent neural control", *Proceedings of the 16th IFAC World Congress*, Praga, Czech Republic, 2005.

[11] S. S. Ge, J. Zhang, and T. H. Lee, "Adaptive neural network control for a class of MIMO nonlinear systems with disturbances in discrete-time", *IEEE Transactions on Systems, Man and Cybernetics*, Part B, vol. 34, no. 4, pp. 1630-1645, 2004.

[12] R. Grover and P. Y. Hwang, *Introduction to Random Signals and Applied Kalman Filtering*, 2nd ed., John Wiley and Sons, New York, N.Y., USA, 1992.

[13] S. Haykin, *Kalman Filtering and Neural Networks,* John Wiley and Sons, New York, N.Y., USA, 2001.

[14] S. Haykin, *Neural Networks, a Comprehensive Foundation*, 2nd ed., Prentice Hall, Upper Saddle River, N.J., USA, 1991.

[15] P. A. Ioannou and J. Sung, *Robust Adaptive Control*, Prentice Hall, Upper Saddle River, N.J., USA, 1996.

[16] N. Kazantzis and C. Kravaris, "Time discretization of nonlinear control systems via Taylor methods", *Computer and Chemical Engineering*, vol. 23, pp. 763-784, 1999.

[17] A. Kapoor, N. Simaan, and P. Kazanzides, "A system for speed and torque control of DC motors with application to small snake robots" *Proceedings of the IEEE Mechatronics & Robotics*, Aachen, Germany, 2004.

[18] H. Khalil, *Nonlinear Systems*, 2nd ed., Prentice Hall, Upper Saddle River, N. J., USA, 1996.

[19] E. B. Kosmatopoulus, A. Manolis, and P. A. Ioannou, "Dynamical neural networks that ensure exponential identification error convergence" *Neural Networks*, vol. 10, pp. 299-314, 1997.

[20] P. C. Krause, O. Wasynczuk, and S. D. Sudhoff, *Analysis of Electric Machinery and Drive Systems*, 2nd ed., John Wiley and Sons, New York, N.Y., USA, 1989.

[21] LEM Components, "Application notes, Current Transducer HX-10-P" Datasheet (http://www.lemusa.com), Milwaukee, WI., USA, 2005.

[22] W. Lin. and C. I. Byrnes, "Design of discrete-time nonlinear control systems via smooth feedback" *IEEE Transactions on Automatic Control*, vol. 39, no. 11, pp. 2340-2346, 1994.

[23] J. Linares-Flores and H. Sira-Ramírez, "DC motor velocity control through a DC-to-DC power converter" *Proceedings of the 43rd IEEE Conference on Decision and Control*, Atlantis Paradise Island, Bahamas, pp. 5297-5302, 2004.

[24] G. P. Liu, *Nonlinear Identification and Control, A Neural Approach, (Advances in industrial control)*, Springer-Verlag, London, Great Britain, 2001.

[25] A. G. Loukianov, "A block method of synthesis of nonlinear systems at sliding modes" *Automation and Remote Control*, vol. 59, no. 7 pp. 916-933, 1998.

[26] A. G. Loukianov, J. Rivera, and J. M. Cañedo, "Discrete-time sliding mode control of an induction motor", *Proceedings of the 15th IFAC World Congress*, Barcelona, Spain, 2002.

[27] A. G. Loukianov, J. M. Cañedo, V. Utkin, and J. Cabrera-Vázquez, "Discontinuous controller for power systems: sliding-mode block control approach" *IEEE Transactions on Industrial Electronics*, vol. 51, no. 2, pp. 340-353, Apr. 2004.

[28] A. G. Loukianov and V. I. Utkin, "Methods of reducing equations for dynamic systems to a regular form" *Automation and Remote Control,* vol. 42, no. 4 pp. 413-420, 1981.

[29] A. G. Loukianov, "Robust block decomposition sliding mode control design", *Mathematical Problems in Engineering*, Taylor & Francis, vol. 8(4-5), pp. 349-365, 2002.

[30] R. Marino and P. Tomei, "Adaptive tracking and disturbance rejection for uncertain nonlinear systems" *IEEE Transactions on Automatic Control*, vol. 50, no. 1, pp. 90-95, 2005.

[31] R. Marino and P. Tomei, "Output regulation for linear systems via adaptive internal model" *IEEE Transactions on Automatic Control*, vol. 48, no. 12, pp. 2199-2202, 2003.

[32] The MathWorks, Inc., "Simulink, Simulation and Model-Based Design" Matlab 2007A, The Language for Technical Computing, Natick, MA., USA, 2007.

[33] G. Obregón-Pulido, B. Castillo-Toledo, and A. G. Loukianov, "Adaptive internal model regulation of a PVTOL vehicle" *Proceedings of the 2nd Symposium on Systems, Structure and Control*, Oaxaca, Oaxaca, México, 2004.

[34] T. Poggio and F. Girosi, "Network for approximation and learning" *Proceedings of the IEEE*, vol. 78, pp. 1481-1497, 1989.

[35] G. A. Rovithakis and M. A. Chistodolou, *Adaptive Control with Recurrent High-Order Neural Networks*, Springer-Verlag, New York, N.Y., USA, 2000.

[36] G. J. Rubio, A. G. Loukianov, J. M. Cañedo, and Y. Vershinin, "Observer-based sliding mode block control of DC motors with controlled excitation" *Conference T&D*, Brasil, 2002.

[37] E. N. Sanchez, A. Y. Alanis, and G. Chen, "Recurrent neural networks trained with Kalman filtering for discrete chaos reconstruction", *Proceedings of Asian-Pacific Workshop on Chaos Control and Synchronization*, Melburne, Australia, 2004.

[38] Semikron, "Application Note: IGBT Power Electronics Teaching System" Principle for sizing power converters (http://www.semikron.com), Nuremberg, B., Germany, 2005.

[39] P. C. Sen, *Principles of Electric Machines and Power Electronics*, 2nd ed., John Wiley and Sons, New York, N.Y., USA, 1997.

[40] A. Serrani, A. Isidori, and L. Marconi, "Semiglobal nonlinear output regulation with adaptive internal model" *IEEE Transactions on Automatic Control*, vol. 46, no. 8, pp. 1178-1194, 2001.

[41] S. Singhal and L. Wu, "Training multilayer perceptrons with the extended Kalman algorithm" *Advances in Neural Information Processing Systems 1*, San Mateo, CA., USA, pp. 133-140, 1989.

[42] Y. Song and J. W. Grizzle, "The extended Kalman filter as local asymptotic observer for discrete-time nonlinear systems" *Journal of Mathematical systems, Estimation and Control*, vol. 5, no. 1, pp. 59-78, Birkhauser-Boston, 1995.

[43] Staco Energy, "Variable Autotransformers, Intalation & Operation Instructions" Staco Energy Products Co. (http://www.staco.com), Dayton, OH., USA, 2007.

[44] V. Utkin, J. Guldner, and J. Shi, *Sliding Mode Control in Electromechanical Systems*, Taylor & Francis, London, United Kingdom, 1999.

[45] W. F. Xie, "Sliding-mode-observer-based adaptive control for servo actuator with friction", *IEEE Transactions on Industrial Electronics*, vol. 54, no. 3, pp. 1517–1527, 2007.

[46] W. C. Su, S. V. Drakunov, Ü. Özgüner, and K. D. Young, *Implementation of Variable Structure Control for Sampled Data Systems*, Springer-Verlag, Berlin, Germany, 1996.

[47] B. Widrow, D. Rumelhart, and M. Lehr, "Neural networks: applications in industry, business and science" *Communications of the ACM*, March, vol. 7, no. 3, 1987.

[48] L. Yu-Feng and J. Wikander, "Discrete-time sliding mode control for linear systems with nonlinear friction" *Proceedings of the 6th IEEE International Workshop on Variable Structure Systems*, pp. 35-44, Gold Coast, Australia, 2000.

Appendix A

Block Linearizing Transformation

This appendix presents the procedure of reducing the system (2.36) to the NBC form (2.40) and close follows [25], [26] and [29].

In this dissertation, system (2.40) is considered with structure $n_1 = n_2 = \ldots = n_r = m$, where the block control design procedure consists of a step-by-step construction of a new system with states $\varepsilon_j(k) = x_j(k) - x_{j,d}(k)$, $j = 1, \ldots, r$, where $x_{j,d}(k)$ is the desired value for $x_j(k)$, which will be defined by such construction.

Define a new variable, $\varepsilon_1(k) = e(k)$, (see (2.38)), i.e.,

$$\varepsilon_1(k) = x_1(k) - x_{1,d}(k) \tag{A.1}$$

with $x_{1,d}(k) = g(\omega(k))$. Once defined the new first variable (A.1), one step ahead is done as

$$\varepsilon_1(k+1) = f_1(x_1(k)) + B_1(x_1(k))x_2(k) + \overline{d}_1(\omega(k)) \tag{A.2}$$

where $\overline{d}_1(\omega(k)) = d_1(\omega(k)) - x_{1,d}(k+1)(\omega(k))$.

Equation (A.2) is viewed as a block with state $\varepsilon_1(k)$, and $x_2(k)$ as pseudo-control input, where desired dynamics must to be imposed. This can be solved using Assumption 1 along with the anticipation of the desired dynamics for this block as follows:

$$\varepsilon_1(k+1) = f_1(x_1(k)) + B_1(x_1(k))x_2(k) + \overline{d}_1(\omega(k)) = \mathbf{K}_1\varepsilon_1(k) \tag{A.3}$$

where the design matrix $\mathbf{K}_1$ is chosen as $\mathbf{K}_1 = diag\{\mathbf{k}_{11}, \ldots, \mathbf{k}_{1m}\}$ with $|\mathbf{k}_{1q}| < 1, \quad q = 1, \ldots, m$ to assure stability of block (A.3). From (A.3), $x_2(k)$ is calculated as

$$x_2(k) = [B_1(x_1(k))]^{-1}\left(\mathbf{K}_1\varepsilon_1(k) - f_1(x_1(k)) - \overline{d}_1(\omega(k))\right). \tag{A.4}$$

Note that the calculated value of state $x_2(k)$ in (A.4) is not the real value of such state; instead, it represents the desired behavior for $x_2(k)$ that drives the first block to

the desired dynamics. In order to avoid confusions, this desired value of $x_2(k)$ is redefined as $x_{2,d}(k)$, so, equation (A.4) is rewritten as

$$x_2(k) = [B_1(x_1(k))]^{-1} \left(\mathbf{K}_1\varepsilon_1(k) - f_1(x_1(k)) - \overline{d}_1(\omega(k))\right)$$

Proceeding in the same way as for the first block, a second variable in new coordinates is defined, $\varepsilon_2(k) = x_2(k) - x_{2,d}(k)$. Taking one step ahead in $\varepsilon_2(k)$ yields

$$\begin{aligned}\varepsilon_2(k+1) &= x_2(k+1) - x_{2,d}(k+1) \\ &= f_2(x_1(k), x_2(k)) + B_2(x_1(k), x_2(k))x_3(k) + \overline{d}_2(\omega(k))\end{aligned}$$

where $\overline{d}_2(\omega(k)) = d_2(\omega(k)) - x_{2,d}(k+1)$. The desired dynamics for this block are imposed as follows:

$$\varepsilon_2(k+1) = f_2(x_1(k), x_2(k)) + B_2(x_1(k), x_2(k))x_3(k) + \overline{d}_2(\omega(k)) = \mathbf{K}_2\varepsilon_2(k) \qquad \text{(A.5)}$$

where $\mathbf{K}_2 = diag\{\mathbf{k}_{21}, \ldots, \mathbf{k}_{2m}\}$ with $|\mathbf{k}_{2q}| < 1, \quad q = 1, \ldots, m$. Using Assumption 1, the desired value $x_{3,d}(k)$ is calculated from (A.5) as

$$x_{3,d}(k) = [B_2(x_1(k), x_2(k))]^{-1} \left(\mathbf{K}_2\varepsilon_2(k) - f_2(x_1(k), x_2(k)) - \overline{d}_2(\omega(k))\right).$$

These steps are taken iteratively. At the last step, the known desired variable is $x_{r,d,k}$, and the last new variable is defined as $\varepsilon_r(k) = x_r(k) - x_{r,d}(k)$. As usually, taking one step ahead, we have

$$\varepsilon_r(k+1) = f_r(x_1(k), \ldots, x_r(k)) + B_r(x_1(k), \ldots, x_r(k))u(k) + \overline{d}_r(\omega(k))$$

where $\overline{d}_r(\omega(k)) = d_r(\omega(k)) - x_{r,d}(k+1)$. It is worth to mention that the new variables $\varepsilon_j(k)$, $j = 1, \ldots, r$, compound a nonlinear transformation of the state $x(k)$, defined as

$$\begin{aligned}
\varepsilon_1(k) &= x_1(k) - x_{1,d}(k) = \varphi_1(x_1(k), \omega(k)) \\
\varepsilon_2(k) &= x_2(k) - [B_1(x_1(k))]^{-1} \left(\mathbf{K}_1\varepsilon_1(k) - f_1(x_1(k)) - \overline{d}_1(\omega(k))\right) \\
&= \varphi_2(x_1(k), x_2(k), \omega(k)) \\
\varepsilon_3(k) &= x_3(k) - [B_2(x_1(k), x_2(k))]^{-1} \left(\mathbf{K}_2\varepsilon_2(k) - f_2(x_1(k), x_2(k)) - \overline{d}_2(\omega(k))\right) \\
&= \varphi_3(x_1(k), x_2(k), x_3(k), \omega(k)) \\
&\vdots \\
\varepsilon_r(k) &= x_r(k) - x_{r,d}(k) = \varphi_r(x_1(k), x_2(k), \ldots, x_r(k), \omega(k)).
\end{aligned}$$

In this way, a diffeomorphic transformation $\varepsilon(k) = \varphi(x(k), \omega(k))=[\varphi_1^T \dots \varphi_r^T]^T$ is defined, simplifying system (2.40) to the following form:

$$
\begin{aligned}
\varepsilon_1(k+1) &= \mathbf{K}_1\varepsilon_1(k) + B_1\varepsilon_2(k) \\
\varepsilon_2(k+1) &= \mathbf{K}_2\varepsilon_2(k) + B_2\varepsilon_3(k) \\
&\vdots \\
\varepsilon_{r-1}(k+1) &= \mathbf{K}_{r-1}\varepsilon_{r-1}(k) + B_{r-1}\varepsilon_r(k) \\
\varepsilon_r(k+1) &= f_r(x_1(k), \dots, x_r(k)) + B_r(x_1(k), \dots, x_r(k))u(k) + \bar{d}_r(\omega(k)).
\end{aligned}
\tag{A.6}
$$

Appendix B

Publications

This sections displays the publications presented during the present project.

B.1 Journal Paper (Submitted for revision)

- C. E. Castañeda, E. N. Sánchez, A. G. Loukianov, and B. Castillo-Toledo, "Real-Time Discrete Recurrent High Order Neural Control for a DC Motor" *IEEE Transactions on Industrial Electronics.*

B.2 Conference Papers

- C. E. Castañeda, E. N. Sánchez, A. G. Loukianov, and B. Castillo-Toledo, "Real-Time DC Motor Torque Control using High Order Neural Networks" *Proceedings of the 3rd IEEE Multi-conference on Systems and Control*, Saint Petersburg, Russia, 2009.

- C. E. Castañeda, E. N. Sánchez, A. G. Loukianov, and B. Castillo-Toledo, "Real-Time DC Motor Control based on Recurrent Neural Networks" *Proceedings of the XIII Congreso Latinoamericano de Control Automático VI Congreso Venezolano de Automatización y Control*, Mérida, Venezuela, 2008.

- C. E. Castañeda, E. N. Sánchez, A. G. Loukianov, and B. Castillo-Toledo, "Real-Time Discrete Neural Control for a DC Motor" *Proceedings of the IEEE Multi-conference on Systems and Control*, San Antonio, Texas, USA, 2008.

- C. E. Castañeda, E. N. Sánchez, A. G. Loukianov, and B. Castillo-Toledo, "Discrete-Time Recurrent Neural Dc Motor Control using Kalman Learning" *Proceedings of*

the IEEE World Congress on Computational Intelligence, Hong Kong, 2008.

El Jurado designado por la Unidad Guadalajara del Centro de Investigación y de Estudios Avanzados del Instituto Politécnico Nacional aprobó la tesis

Control de un Motor de Corriente Directa basado en Redes Neuronales Recurrentes

del (la) C.

Carlos Eduardo CASTAÑEDA HERNÁNDEZ

el día 28 de Agosto de 2009.

Dr. Edgar Nelson Sánchez Camperos
Investigador CINVESTAV 3D
CINVESTAV Unidad Guadalajara

Dr. Bernardino Castillo Toledo
Investigador CINVESTAV 3C
CINVESTAV Unidad Guadalajara

Dr. Juan Manuel Ramírez Arredondo
Investigador CINVESTAV 3C
CINVESTAV Unidad Guadalajara

Dr. Alexander Georgievich Loukianov
Investigador CINVESTAV 3C
CINVESTAV Unidad Guadalajara

Dr. Arturo Román Messina
Investigador CINVESTAV 3C
CINVESTAV Unidad Guadalajara

Dra. Alma Yolanda Alanis García
Profesor
CUCEI UDG