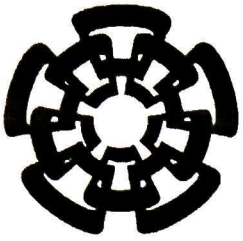
xx (113534.1)



CINVESTAV

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

Control Neuronal de Estructura Variable

Tesis que presenta:
Ramón Antonio Félix Cuadras

para obtener el grado de:
Doctor en Ciencias

en la especialidad de:
Ingeniería Eléctrica

Guadalajara, Jal., Agosto del 2003.

Control Neuronal de Estructura Variable

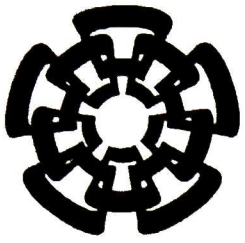
Tesis de Doctorado en Ciencias
Ingeniería Eléctrica

Por:
Ramón Antonio Félix Cuadras
Ingeniero Electrónico.
Instituto Tecnológico de Mar Campus Mazatlán
1994-1998
Maestro en Ciencias con especialidad en Ingeniería Eléctrica. CINVESTAV del IPN Unidad Guadalajara
1998-2000

Becario del CONACyT, expediente no. 129303

Directores de Tesis
Dr. Edgar Nelson Sánchez Camperos
Dr. Alexander Georgievich Loukianov.

CINVESTAV del IPN Unidad Guadalajara, Agosto del 2003



CINVESTAV

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

Variable Structure Neural Control

Thesis submitted by:
Ramón Antonio Félix Cuadras

For the degree of:
Doctor on Sciences

On specialty of:
Electrical Engineering

Guadalajara, Jal., August 2003.

Variable Structure Neural Control

Doctor on Sciences
Electrical Engineering

By:
Ramón Antonio Félix Cuadras
Engineering on Electronics.
Instituto Tecnológico de Mar Campus Mazatlán
1994-1998
Master on Sciences with specialty on Electrical Engineering. CINVESTAV del IPN Unidad Guadalajara
1998-2000

CONACyT file no. **129303**

Thesis advisors
Dr. Edgar Nelson Sánchez Camperos
Dr. Alexander Georgievich Loukianov.

CINVESTAV del IPN Unidad Guadalajara, August 2003

Agradecimientos:

A mis padres y hermanos por haber sido una fuente de inspiración durante mis estudios.

A mis asesores Dr. Edgar Sánchez y Dr. Alexander Luokianov por sus enseñanzas, esfuerzo y paciencia para sacar adelante este trabajo de investigación.

A los miembros del jurado de tesis; Dr. Bernardino Castillo, Dr. Juan Manuel Ramírez, Dr. Gerardo Espinosa y Dr. Iuori Orlov, por sus valiosos comentarios que ayudaron a mejorar esta tesis.

Al CINVESTAV Guadalajara, por permitirme realizar estos estudios.

Al CONACYT por apoyarme económicamente durante estos años.

Al Dr. Romeo Ortega por invitarme a formar parte de su equipo de trabajo durante mi estancia doctoral en LSS-SUPELÉC.

A los amigos y compañeros que me ayudaron y apoyaron en los momentos difíciles durante mi estancia en Francia: Chava, William, Martha, Victor, Eloisa, Marcelo, Figth, Lee, Raphaelle y David

A mis amigos y compañeros del CINVESTAV por haber compartido los momentos buenos y malos durante estos 5 años. Especialmente agradezco al Trofeo por sus nuevas palabras de aliento, Al Droopy por el diseño de sus paginas personales de internet, al Chido por sus paros chidezcos, al Reza por el pan y finalmente al Pipo y al Lobo por haberme enseñado a jugar el Risk.

Resumen

La tecnología en redes neuronales ha llamado mucho la atención en años recientes; su habilidad para aprender relaciones no lineales es ampliamente apreciada y se utiliza en diferentes aplicaciones: el modelado de sistemas dinámicos, procesamiento de señales y diseño de sistemas de control son de las más comunes. La teoría de neurocomputación ha madurado considerablemente durante la pasada década y muchos problemas de redes neuronales: diseño, entrenamiento y evaluación han sido resueltos.

La teoría de sistemas de estructura variable ha mostrado un enorme potencial en varios campos de investigación. En particular, el control de estructura variable ha probado ser una valiosa herramienta para diseñar sistemas de control, debido, entre otras cosas, a la robustez contra las perturbaciones que cumplan la llamada condición de acoplamiento. Otra característica del control de estructura variable es la descomposición del diseño del controlador en dos pasos; primero se selecciona una variedad deslizante donde el sistema se comporta de una manera adecuada, y segundo se diseña un control discontinuo que lleve al sistema a dicha variedad y lo mantenga ahí.

El objetivo principal de esta tesis es proponer un esquema de control basado en identificación por redes neuronales y en sistemas de estructura variable. Dicho esquema es aplicable para un amplio conjunto de sistemas no lineales, llamados sistemas controlables a bloques. Una red neuronal recurrente se usa para identificar la planta, con una ley de adaptación de pesos que garantiza esta tarea. Basado en este identificador neuronal, se diseña el controlador de modos deslizantes que lleva el estado del identificador a la variedad deslizante deseada; dicha variedad se diseña usando la técnica de control a bloques. El sistema completo de control garantiza que la salida del sistema real siga a una señal de referencia pre-especificada. Adicionalmente, esta estrategia de control permite usar identificadores de orden parcial; esto reduce considerablemente la complejidad computacional del controlador.

La aplicabilidad de este esquema de control se prueba en un motor de inducción y en un generador síncrono, por medio de simulaciones. El control neuronal a bloques para el motor de inducción muestra gran robustez contra las perturbaciones de par de carga y las variaciones en la resistencia del rotor. Además, se derivaron dos leyes de control alternativas para este motor: un controlador de modos deslizantes singulares y un controlador neuronal a bloques con restricciones de entrada. Ambos presentan un desempeño muy similar al mostrado por el controllador neuronal a bloques original. Por otro lado, el controlador del generador síncrono se diseñó haciendo ciertos cambios al controlador neuronal propuesto anteriormente. Sin embargo, esta nueva estrategia de control muestra gran robustez bajo las perturbaciones de corto circuito en las barras terminales.

Finalmente, se presenta un nuevo tópico; las redes neuronales recurrentes de estructura variable. La teoría de sistemas de estructura variable se utiliza para analizar el comportamiento de estas redes. El objetivo de este esquema de identificación es seleccionar una estructura adecuada para el identificador neuronal. Los resultados obtenidos en simulacion son muy prometedores cuando se aplica este esquema a la identificación de sistemas caóticos; en particular el esquema se prueba en el sistema caótico de Chen y en el circuito caótico de Chua.

Abstract

The technology of neural networks has attracted much attention in recent years; their ability to learn nonlinear relationships is widely appreciated and are used in many different classes of applications; modelling of dynamics systems, signal processing and control-system design being some of the most common. The theory of neural computing has matured considerably over the last decade and many problems of neural-network; design, training and evaluation have been yet solved.

The Variable Structure Systems (VSS) theory has revealed an enormous potential in several interesting research trends. In particular, Variable Structure Control (VSC) has become an effective tool for control systems design, due to, among other things, the robustness in presence of disturbances that satisfy the so-called matching condition. Other feature of Variable Structure Control is the decomposition of the controller design in two steps: first, the selection of the sliding manifold where the system exhibits a well-behavior, and second. the design of a discontinuous control law which drives the system into the desired sliding manifold and keep it there.

The main goal of this thesis is to propose a control scheme based on neural networks identification and Variable Structure Systems. Such scheme is applicable to a wide class of nonlinear systems called Block Controllable (BC) systems. A recurrent neural network is used to identify the plant, with an update law that guarantees such task. The sliding modes controller is designed to drive the identifier state into the desired sliding manifold, which is designed using the Block Control technique. The overall control system guarantees that the output of the real system tracks a pre-specified reference signal. Additionally, this strategy allows to use partial state identifiers; this fact reduce considerably the computational complexity of the Neural Block Controller (NBC).

The robustness of this control scheme is tested on an induction motor and a synchronous generator. The Neural Block Controller for induction motors shows to be robust respect to load torque and rotor resistance variations. Additionally, two alternative control laws are derived to control this motor: Singular Sliding Mode (SSM) controller and the Neural Block Control with input constrains. Both controllers present a performance very similar as the Neural Block Controller proposed originally. On the other hand, the synchronous generator controller was designed doing slight changes to the NBC, in order to fit this scheme to the plant model. Nevertheless, this new control strategy shows to be robust respect to short circuit disturbances. Simulations results are presented for both systems.

Finally, a new topic is presented; the Variable Structure Recurrent Neural Networks (VSRNN) for nonlinear system identification. VSS theory is used to analyze the VSRNN behavior. The aim of this identification scheme is to select an adequate neural identifier structure. The simulation results are very encouraging when the scheme is applied to chaos identification. In particular the scheme is tested with the Chen's chaotic system and the Chua's chaotic circuit.

Contents

Chapter 1

Introduction

Artificial Neural Networks (ANN) are systems inspired from how the human brain works. The brain is a highly complex, nonlinear, and parallel information-processing system. It has the capability of organizing its structural elements, called neurons, which interact massively among them, through synaptic connections. So, the biological neural networks are capable to perform certain tasks (e.g. pattern recognition, perception, motion control, etc.) in approximately 100-200 ms, whereas tasks of much less complexity may take days on a conventional computer [19].

An ANN consists 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 neural networks is its ability to learn from its environment, and to improve its performance through learning [46]. The performance improvement takes place over time in accordance with some prescribed measure. A neural network learn about its environment by an active process of adjustment applied to its synaptic weights. Ideally, the network becomes more knowledgeable about its environment as the learning process goes through.

Other basic feature of neural architectures is that they work in parallel. Although ANNs can perform human brain-like tasks such as object and pattern recognition or associative memory, there is still a big distance between them and the biological ones. Nevertheless, they are certainly a powerful tool to deal with a large set of interesting problems. Indeed, ANNs have provided good solutions to many problems in various fields: such as classification, vision, speech, signal processing, time series prediction, modelling and control, robotics, optimization, experts systems and financial applications, among others [62].

Several motives have originally leaded researchers to study neural networks. One of the primary motives was to create a computer program able to learn from experience. When the experience mentioned is interpreted as knowledge about how certain inputs affecting a plant, it

is obvious that neural networks must have something in common with the techniques applied in control systems.

1.1 Neural Networks in Control Systems

Frequently, modern control systems require a very structured knowledge about the system to be controlled; such knowledge should be represented in terms of differential or difference equations. This mathematical description of the dynamic system is named as the model. Basically, there are two ways to obtain a model; it can be derived in a deductive manner using physics laws, or it can be inferred from a set of data collected during a practical experiment with the system. The first method can be simple, but in many cases it is excessively time-consuming. Sometimes, it may be even considered unrealistic or impossible to obtain an accurate model in this way. The second method, which is commonly referred as identification system, it could be a useful short cut for deriving mathematical models. Although system identification not always results in equally accurate model, a satisfactory model can often be obtained with reasonable effort. The main drawback is the requirement to conduct a practical experiment which brings the system through its range of operation. Also a certain knowledge about the plant is still required.

System identification are widely used in relation to control systems design and many successful applications have been made over the years. Sometimes system identification is even implemented as an integral part of the controller. This is known as an adaptive controller and it is typically designed to control systems whose dynamical characteristics vary with time. In the typical adaptive controller a model that is valid under the current operating conditions is identified on-line, and the controller is then, in such way, redesigned in agreement with this model.

Much literature is available on system identification, adaptive control and control system design in general, but traditionally most of it has focused on dealing with models and controllers described by linear differential or difference equations. However, motivated by the fact that all systems exhibit some kind of nonlinear behavior. Recently, there has been much focus on different approaches to nonlinear system identification and controller design. One of the key players in this endeavor are the ANNs. The following features of ANNs makes them particularly attractive for application to modelling and control of nonlinear systems [21]:

- Artificial Neural Networks are universal approximators [8]. It has been proven that any continuous nonlinear function can be approximated arbitrarily well over a compact set by a multilayer neural network which consist of one or more hidden layers.
- Learning and adaptation. The intelligence of neural networks comes from their generalization ability with respect to unknown data. On-line adaptation of the weights is

possible.

- Multivariable systems. ANNs have many inputs and outputs, which makes it easy to model multivariable systems.

Hence, unknown nonlinear functions in dynamical models and controllers can be parametrized by means of neural networks architectures. Although this fact has enormous potential abilities, there are also a number of weak points, for example: the existence of many local optima in learning algorithms, the choice of complexity of the neural networks and the stability analysis of dynamical systems which contains ANN architectures.

Most of the applications of neural networks to nonlinear identification and control is based on the feedforward ones [18], [21]. Lately, the use of recurrent neural networks which allows a more efficient modeling of dynamic systems, is increasing [57], [47]. A recurrent neural network distinguishes itself from feedforward ones because it has at least one feedback loop. For example, a recurrent neural 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 [19].

In this research, the neural identifier is built using High Order Recurrent Neural Networks (RHONN) [28]; which are an efficient tool for nonlinear identification and ensures error exponential convergence [29]. However, such algorithm requires too much computations, so a simpler adaptation parameters algorithm is used [48].

1.2 Variable Structure Control

Usually, the Variable Structure Control (VSC) algorithms [59] enforce sliding mode motion into some manifold of the state space. The methodology for sliding mode control consists basically of two steps. Sliding motion is governed by a reduced order system depending on the equations of some surfaces, whose intersection describes a manifold, which is called the sliding manifold.

The first stage of design is the selection of the sliding manifold where the sliding motion exhibits desired properties. For this stage, many standard control methods can be applied; stabilization, pole replacement, dynamic optimization, etc. In this research work, the so-called Nonlinear Block Control technique [34] is used to design such sliding manifold.

The second stage is to find a discontinuous control law which enforces sliding mode on the sliding manifold selected at the first stage. This second problem is of reduced order as well, since its dimension is equal to the number of discontinuity surfaces, which is usually equal to the dimension of the control space.

By separating the control problem in two motions of lower dimensions – the first motion proceeds sliding mode within a finite time interval and the second motion is the sliding mode with the desired properties –, the design procedure is considerably simplified. Additionally, sliding modes are insensitive with respect to unknown plant parameters and external disturbances which satisfies the so-called matching condition.

This thesis proposes a trajectory tracking control scheme which combines VSC and neural networks approach [52][36][12]. So that, the matched disturbances are rejected by VSC and the effect of unmatched ones is compensated by neural identification. Modifying existing identification schemes based on recurrent neural networks [28], a neural network identifier of block controllable form is proposed. Based on this model, a discontinuous control law which combines block control [34] and VSC with sliding mode techniques [58], is derived. The block control approach is used to design a nonlinear sliding surface such that the resulting sliding mode dynamics is described by a desired linear system. The proposed neural identifier and control strategy allow trajectory tracking for systems which are represented in the nonlinear block controllable form.

1.3 Thesis Outline

The outline of this thesis is as follows:

- Chapter 2. The neural model used through this work is introduced. This model, called Recurrent High Order Neural Networks (RHONN), is a polynomial extension of the well-known Hopfield neural network.
- Chapter 3. The neural block control is explained for a the block controllable systems; a particular case when the relative degree equal or less than 2, is analyzed in detail.
- Chapter 4. Three control strategies are derived for induction motor control application; Neural Block Control (NBC), Singular Sliding Modes (SSM) control and NBC with input restrictions. The results are published in [36] and [13].
- Chapter 5. A control law is derived to reject the effect of the short circuit disturbance in a synchronous generator connected to a infinite bus; a three-order model is used to simulate this generator. [12] resumes the obtained results.
- Chapter 6. Based on RHONN identifiers, the nonlinear systems identification via Variable Structure Recurrent Neural Network (VSRNN) is proposed. This scheme is not only a parametric identification but also a structural identification. [49] and [50] are fruits of this research.

- Chapter 7. Finally, the conclusions of this work are presented.
- Appendix A. The block control with input constrains for inductions motors is explained.

Chapter 2

Recurrent High-Order Neural Networks

The use of multilayer neural networks is well known for pattern recognition and for modelling of static systems. The network is trained to learn an input-output map [46]. Theoretical works have proven that, even with one hidden layer, a neural network can uniformly approximate any continuous function over a compact domain, provided that the network has a sufficient number of synaptic connections.

A typical nonlinear identification problem consists of selecting an appropriate model and adjusting its parameters according with some adaptive law, such that the response of the model to an input signal approximates the response of the real system.

Depending on the level of *a priori* knowledge about the plant, the identification problem can be approached in different ways. If the identification is based exclusively on measured data, assuming no knowledge about the physics of the plant, the identification process is called *black-box modelling*, which is the typical approach used for neural networks identification schemes. In contrast to this, the *white-box modelling* is used for a pure physical modelling of the system. When certain level of insight about the system exists and is utilized to improve the empirical modelling, the phrase *gray-box modelling* is used. This is the approach used in this work; we take some information about the plant structure and use it to select the identifier structure. In particular, the block controllable form and the relative degree are taken into account to build the neural identifier.

2.1 Neural Model

For the identification task, expansions of the first order Hopfield model called High Order Recurrent Neural Networks (RHONN) are used [28], which presents more interactions among the neurons [29]. Additionally, the RHONN model is very flexible and allows to incorporate to the neural identifier *a priori* information about the plant structure.

A recurrent high-order recurrent neural network of n neuron and m inputs is defined as [29]

$$\dot{x}_i = -a_i x_i + \sum_{k=1}^{L_i} w_{ik} \prod_{j \in I_k} \eta_k^{d_j(k)}, \qquad i = 1, ..., n \tag{2.1}$$

where x_i is the i-th neuron state, L_i is the number of high order connections, $\{I_1, I_2, ..., I_L\}$ is a collection of non-ordered subsets of $\{1, 2, ..., m+n\}$, $a_i > 0$, w_{ik} are the adjustable weights of the neural network, $d_j(k)$ are non-negative integers, and η is a vector defined as

$$\boldsymbol{\eta} = \begin{bmatrix} \eta_1 \\ \vdots \\ \eta_n \\ \eta_{(n+1)} \\ \vdots \\ \eta_{(n+m)} \end{bmatrix} = \begin{bmatrix} S(x_1) \\ \vdots \\ S(x_n) \\ u_1 \\ \vdots \\ u_m \end{bmatrix}$$

with $u = [u_1, u_2, ..., u_m]^T$ being the input to the neural networks, and $S(\cdot)$ a smooth hyperbolic tangent function formulated by

$$S(x) = \frac{2}{1 + \exp(-\beta x)} - 1.$$

Hence $S(x) \in [-1, 1]$. As can be seen, (2.1) allows the inclusion of high order terms.

Let define the vector

$$\boldsymbol{\rho}_i(\mathbf{x}, \mathbf{u}) = \begin{bmatrix} \rho_{i,1} \\ \rho_{i,2} \\ \vdots \\ \rho_{i,L_i} \end{bmatrix} = \begin{bmatrix} \prod_{j \in I_1} \eta_j^{d_j(1)} \\ \prod_{j \in I_2} \eta_j^{d_j(2)} \\ \vdots \\ \prod_{j \in I_L} \eta_j^{d_j(L_i)} \end{bmatrix}$$

It is worth mentioning that the entries of $\rho_i(x, u)$ are multiplying combination of elements of $\eta(x, u)$. In a more compact notation, (2.1) can be rewritten as

$$\dot{x}_i = -a_i x_i + \sum_{k=1}^{L_i} w_{ik} \rho_{ik}, \qquad i = 1, ..., n$$

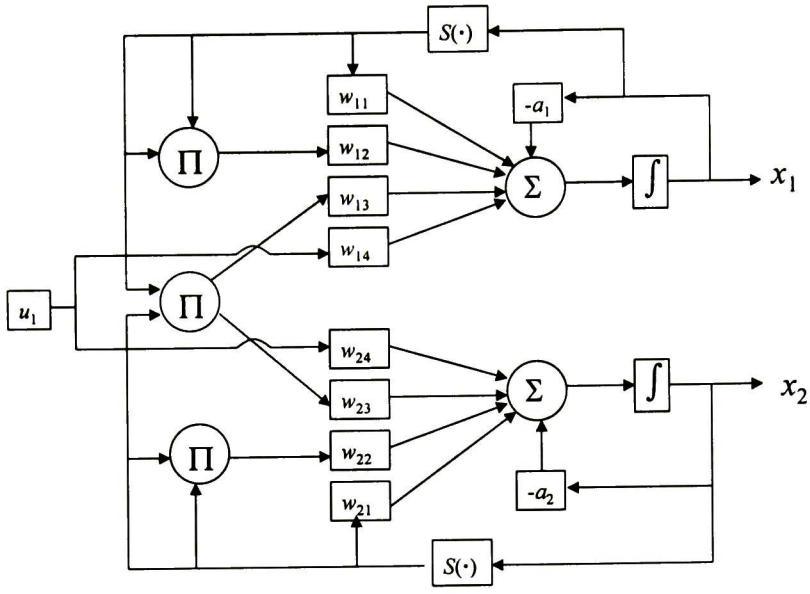


Figure 2.1: A 2nd-order RHONN example.

or

$$\dot{x}_i = -a_i x_i + \mathbf{w}_i^\top \boldsymbol{\rho}_i(\mathbf{x}, \mathbf{u}), \qquad i = 1, ..., n \tag{2.2}$$

where $\mathbf{w}_i = [w_{i,1} ... w_{i,L_i}]^\top$

Figure 2.1 shows a RHONN example, where

$$\begin{aligned} \dot{x}_1 &= -a_1 x_1 + w_{11} S(x_1) + w_{12} S(x_1)^2 + w_{13} S(x_1) S(x_2) + w_{14} u_1 \\ \dot{x}_2 &= -a_2 x_2 + w_{21} S(x_2) + w_{22} S(x_2)^2 + w_{23} S(x_1) S(x_2) + w_{24} u_1 \end{aligned}$$

with $n = 2$, $m = 1$, $L_1 = L_2 = 4$.

2.2 On-line Identification

In this section, we consider the problem of identifying a nonlinear system given by

$$\dot{\boldsymbol{\chi}} = \mathbf{f}(\boldsymbol{\chi}, \mathbf{u}) \tag{2.3}$$

where $\boldsymbol{\chi} \in \Re^n$, $\mathbf{u} \in \Re^m$, $\mathbf{f}$ is a smooth vector field and $f_i(\boldsymbol{\chi}, \mathbf{u})$ its entries. In order to identify system (2.3), as discussed in [29], we assume that it is fully described by a RHONN, with each

neuron state given by

$$\dot{\chi}_i = -a_i\chi_i + \mathbf{w}_i^{*\top}\boldsymbol{\rho}_i(\boldsymbol{\chi}, \mathbf{u}) + \nu_i(\boldsymbol{\chi}, \mathbf{u}) \tag{2.4}$$

with $\mathbf{w}_i^*$, $\boldsymbol{\rho}_i \in \Re^{L_i}$. The optimal unknown parameters vector $\mathbf{w}_i^*$ is defined as

$$\mathbf{w}_i^* = \arg \min_{\mathbf{w}_i} \left\{ \sup_{\boldsymbol{\chi},\mathbf{u}} |f_i(\boldsymbol{\chi}, \mathbf{u}) + a_i\chi_i - \mathbf{w}_i^{*\top}\boldsymbol{\rho}_i(\boldsymbol{\chi}, \mathbf{u}) - \nu_i(\boldsymbol{\chi}, \mathbf{u})| \right\} \tag{2.5}$$

Let define the modelling error term ν_i as

$$\nu_i(t) = f_i(\boldsymbol{\chi}, \mathbf{u}) + a_i\chi_i - \mathbf{w}_i^{*\top}\boldsymbol{\rho}_i(\boldsymbol{\chi}, \mathbf{u}). \tag{2.6}$$

The proposed neural model (2.3) allows to match the identification algorithm to the real plant.

In order to identify model (2.3), it is assumed that the plant is represented by the proposed RHONN (2.4). Then two possible models can be built.

- Parallel model

$$\dot{x}_i = -a_i x_i + \mathbf{w}_i^{\top}\boldsymbol{\rho}_i(\mathbf{x}, \mathbf{u}), \qquad i = 1, ..., n \tag{2.7}$$

- Series-Parallel model

$$\dot{x}_i = -a_i x_i + \mathbf{w}_i^{\top}\boldsymbol{\rho}_i(\boldsymbol{\chi}, \mathbf{u}), \qquad i = 1, ..., n \tag{2.8}$$

where x_i is the i-th component of the RHONN, and $\boldsymbol{\chi}$ is the plant state. To develop the weight update law, the series-parallel model is used.

2.2.1 On-Line Weight Update Law

Let define the $i-$th identification error

$$e_i = x_i - \chi_i$$

and the $i-$th parameter error

$$\tilde{\mathbf{w}}_i = \mathbf{w}_i - \mathbf{w}_i^*.$$

Assuming that the error modelling term is zero, from (2.3) and (2.4) the identification error dynamics is obtained as

$$\dot{e}_i = -a_i e_i + \tilde{\mathbf{w}}_i^\top \boldsymbol{\rho}_i. \tag{2.9}$$

The next lemma is needed to develop the adaptive law (see Theorem 4.4 in [26]).

Lemma 2.1 *Consider the system*

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) \tag{2.10}$$

where $\mathbf{x} \in \Re^n$ *and* $\mathbf{f}(\mathbf{x})$ *is locally Lipschitz. Assume that there exists a function* $V : \Re^n \rightarrow \Re^+$ *radially unbounded and continuously differentiable such that*

$$\dot{V} = \frac{\partial V}{\partial \mathbf{x}} \mathbf{f}(\mathbf{x}) \leq -W(\mathbf{x}) \leq 0$$

$\forall \mathbf{x} \in \Re^n$, *where* $W(x)$ *is a positive semidefinite function, then all trajectories of (2.10) are bounded for* $t \geq 0$, *moreover*

$$\lim_{t \rightarrow \infty} W(\mathbf{x}(t)) = 0.$$

Using Lemma 2.1 we propose a positive definite function $V_i(e_i, \tilde{\mathbf{w}}_i)$ and a parameter adaptive law such that $\lim_{t \rightarrow \infty} V_i(e_i(t), \tilde{\mathbf{w}}_i(t))$ exists and $\dot{V}_i$ is uniformly continuous and $\dot{V}_i(t) \rightarrow 0$ implies that $e_i(t) \rightarrow 0$, when $t \rightarrow \infty$. Thereafter, let consider the Lyapunov function candidate

$$V_i = \frac{1}{2}(e_i^2 + \tilde{\mathbf{w}}_i^\top \Gamma_i \tilde{\mathbf{w}}_i) \tag{2.11}$$

where Γ_i is a symmetric positive definite matrix. Differentiating (2.11) along the trajectories of (2.9), we obtain

$$\dot{V}_i = -a_i e_i^2 + e_i \tilde{\mathbf{w}}_i^\top \boldsymbol{\rho}_i + \tilde{\mathbf{w}}_i^\top \Gamma_i \dot{\tilde{\mathbf{w}}}_i \tag{2.12}$$

If we define the weight adaptive law [48] as

$$\dot{\mathbf{w}}_i = -e_i \Gamma_i^{-1} \boldsymbol{\rho}_i \tag{2.13}$$

then the equation (2.12) becomes

$$\dot{V}_i = -a_i e_i^2 \leq 0.$$

Using Lemma 2.1, the adaptive law (2.13) ensures that the weights are bounded and the identification error converges to zero, with $\mathbf{x} = [e_i \ \tilde{\mathbf{w}}_i^\top]^\top$ $V(\mathbf{x}) = V_i(e_i, \tilde{\mathbf{w}}_i)$, and $W(\mathbf{x}) = a_i e_i^2$.

2.2.2 Robust Weight Update Law

For the case when the modelling error term is not zero we can guarantee neither the boundness of the parameters nor the convergence to zero of the identification error. Then we need to apply the adaptive law (2.13) with the σ-modification [23] in order to guarantee at least, that the identification error and the weights are bounded for any time. Hence we propose the adaptive law

$$\dot{\mathbf{w}}_i = -\Gamma_i^{-1}(e_i\boldsymbol{\rho}_i - \sigma_i\mathbf{w}_i) \tag{2.14}$$

where σ_i is given as:

$$\sigma_i = \begin{cases} 0, & \text{if } ||\mathbf{w}_i|| \leq M_i \\ \left(\frac{||\mathbf{w}_i||}{M_i}\right)^q \sigma_{i0}, & \text{if } M_i < ||\mathbf{w}_i|| \leq 2M_i \\ \sigma_{i0}, & \text{if } ||\mathbf{w}_i|| > 2M_i \end{cases}$$

with integer $q \geq 1$, and σ_{i_0} and M_i positive constants.

Lemma 2.2 *Assume that the system (2.4) and the RHONN (2.9) and all parameters are adapted using the law (2.14). Then, e_i and w_i converges into a bounded set.*

Proof. The time derivative of V_i (2.11) along the trajectories of (2.9) and (2.14) is given by

$$\dot{V}_i = -a_i e_i^2 - \sigma_i \tilde{\mathbf{w}}_i^\top \mathbf{w}_i - e_i v_i(\boldsymbol{\chi}, \mathbf{u}).$$

Applying the triangular inequality and defining $d_0 = \max_{t\leq 0}(v_i(\boldsymbol{\chi}, \mathbf{u}))$, we have

$$\dot{V}_i \leq -a_i e_i^2 - \sigma_i \tilde{\mathbf{w}}_i^\top \mathbf{w}_i + \frac{e_i^2}{2} + \frac{{d_0}^2}{2}.$$

Since $\tilde{\mathbf{w}}_i = \mathbf{w}_i - \mathbf{w}_i^*$

$$-\tilde{\mathbf{w}}_i^\top \mathbf{w}_i = -(\tilde{\mathbf{w}}_i^\top \tilde{\mathbf{w}}_i + \tilde{\mathbf{w}}_i^\top \mathbf{w}_i^*) \leq -\frac{1}{2}||\tilde{\mathbf{w}}_i||^2 + \frac{1}{2}||\mathbf{w}_i^*||^2$$

therefore

$$\dot{V}_i \leq -a_i e_i^2 + \frac{e_i^2}{2} - \frac{1}{2}\sigma_i||\tilde{\mathbf{w}}_i||^2 + \frac{1}{2}\sigma_i||\mathbf{w}_i^*||^2 + \frac{d_0^2}{2}$$

Selecting $a_i > \frac{1}{2}$, we define $\alpha_i = a_i - \frac{1}{2}$, so that

$$\dot{V}_i \leq -\alpha_i e_i^2 - \frac{1}{2}\sigma_i||\tilde{\mathbf{w}}_i||^2 + \frac{1}{2}\sigma_i||\mathbf{w}_i^*||^2 + \frac{d_0^2}{2}$$

Substituting e_i from (2.11) in the above inequality, we have

$$\dot{V}_i \leq -\alpha V_i + \alpha_i \tilde{\mathbf{w}}_i^\top \Gamma_i \tilde{\mathbf{w}}_i - \frac{1}{2}\sigma_i ||\tilde{\mathbf{w}}_i||^2 + \frac{1}{2}\sigma_i ||\mathbf{w}_i^*||^2 + \frac{d_0^2}{2}.$$

Taking the worst case, when $||\mathbf{w}_i|| > 2M_i$, we select $\sigma_{io} > 2\alpha||\Gamma_i||$, so that

$$\dot{V}_i \leq -\alpha_i V_i + \frac{1}{2}\sigma_{io} ||\mathbf{w}_i^*||^2 + \frac{d_0^2}{2}.$$

Therefore, $[e_i \ \tilde{\mathbf{w}}_i^\top]^\top$ converges on the residual set

$$D_i = \left\{ [e_i \ \tilde{\mathbf{w}}_i^\top]^\top : V_i \leq \frac{1}{\alpha_i} \left(\frac{1}{2}\sigma_{io} ||\mathbf{w}_i^*||^2 + \frac{d_0^2}{2} \right) \right\}$$

and stay there. So the proof is completed. ■

Chapter 3

Neural Block Control

In this scheme, the control law is based on the neural networks, whose inputs are the states of the plant and the RHONN weights are updated depending on the identification error [48]. Figure 3.1 explains the proposed control scheme.

The identification and control scheme used in this research is based on the following property:

Given a desired output trajectory, expressed on output variables as $\mathbf{y}_r$, a nonlinear system with output $\mathbf{y}_P$, and a neural network output $\mathbf{y}_N$, then it is possible to establish the inequality

$$\|\mathbf{y}_r - \mathbf{y}_P\|_2 \leq \|\mathbf{y}_N - \mathbf{y}_P\|_2 + \|\mathbf{y}_r - \mathbf{y}_N\|_2$$

with $\|\cdot\|_2$ as the Euclidean norm.

where $\mathbf{y}_r - \mathbf{y}_P$ is the system output tracking error, $\mathbf{y}_N - \mathbf{y}_P$ is the output identification error and $\mathbf{y}_N - \mathbf{y}_r$ is the RHONN output tracking error. Hence, it is possible to divide the tracking problem in two parts:

1. Minimization of $\|\mathbf{y}_N - \mathbf{y}_P\|_2$,which can be achieved by the proposed on-line identification algorithm.

2. Minimization of $\|\mathbf{y}_N - \mathbf{y}_r\|_2$, for which a tracking algorithm is developed on the basis of the neural identifier (2.8).

The second goal can be reached by designing a control law based on the RHONN model. To design such controller we propose to use of the so called Neural Block Control [52], [51]. This control technique requires the plant to have the Block Controllable Form (BCF) [34], so a RHONN identifier with BCF is proposed. Based on this neural model a discontinuous control

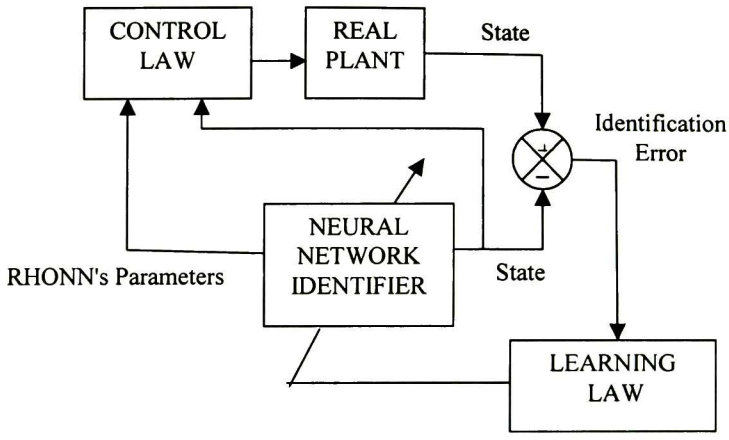


Figure 3.1: Block Control Scheme.

law, which combines block control [34] and VSC with sliding mode technique [58], is derived. The block control approach is used to design a nonlinear sliding surface such that the resulting sliding mode dynamics is described by a desired linear system.

One additional advantage about using VSC is the separation of the system dynamics in two motions, so that, only a partial state RHONN is needed to derive the control law. This strategy is used to control an induction motor in Chapter 4 and a synchronous generator in Chapter 5.

3.1 Nonlinear Block Controllable Form

Let consider the nonlinear system

$$\dot{\chi} = \mathbf{f}(\chi, t) + \mathbf{B}(\chi, t)\,\mathbf{u} \tag{3.1}$$

where $\chi \in D_\chi \subset \Re^n$, and $\mathbf{u} \in \mathbf{U} \subset \Re^m$ is bounded by

$$|\mathbf{u}| \leq u_0, \quad u_0 > 0 \tag{3.2}$$

The vector field $\mathbf{f}(\chi, t)$ and the columns of $\mathbf{B}(\chi, t) = [\ \mathbf{b}_1(\chi, t) \quad \mathbf{b}_2(\chi, t) \quad \cdots \quad \mathbf{b}_m(\chi, t)\ ]$ are sufficiently smooth. Additionally assume that $\mathbf{f}(\mathbf{0}, t) = \mathbf{0}$.

Definition 3.1 *Let the system*

$$\begin{aligned}
\dot{\chi}_1 &= \mathbf{f}_1(\chi_1, t) + \mathbf{B}_1(\chi_1, t)\chi_2 \\
\dot{\chi}_2 &= \mathbf{f}_2(\chi_1, \chi_2, t) + \mathbf{B}_2(\chi_1, \chi_2, t)\chi_3 \\
\dot{\chi}_i &= \mathbf{f}_i(\chi_1, \ldots, \chi_i, t) + \mathbf{B}_i(\chi_1, \ldots, \chi_i, t)_{i+1}, \quad i = 3, \ldots, r-1 \\
\dot{\chi}_r &= \mathbf{f}_r(\chi_1, \ldots, \chi_{r+1}, t) + \mathbf{B}_r(\chi_1, \ldots, \chi_{r+1}, t)\mathbf{u} \\
\dot{\chi}_{r+1} &= \mathbf{f}_{r+1}(\chi_1, \ldots, \chi_{r+1}, t) + \mathbf{B}_{r+1}(\chi_1, \ldots, \chi_{r+1}, t)\mathbf{u}
\end{aligned} \tag{3.3}$$

and the output

$$\mathbf{y}_P = \mathbf{h}(\chi) = \chi_1$$

It is said that this system has the Nonlinear Block Controllable Form with Zero Dynamics, where

$$rank[\mathbf{B}_i] = n_i \quad \forall \chi \in D_\chi \subset \Re^n \text{ and } t \in [0, \infty), \quad i = 1, \ldots, r \tag{3.4}$$

and $\chi^\top = (\chi_1, \chi_2, \ldots, \chi_{r+1})$. *The numbers* $n_1, n_2, \ldots, n_r$ *are known as the controllability indexes and satisfies*

$$n_1 \leq n_2 \leq \cdots \leq n_r \leq m$$

with $\sum_{i=1}^{r+1} n_i = n$.

It is easy to see that the relative degree of the system (3.3) is r with respect to the output $\mathbf{y}_P$

3.2 General Neural Block Controller

Let consider a system with NBC form (3.3) with $n_1 = n_2 = n_3 = ... = n_r$. Additionally, assume that the system has a Block Controllable (BC) RHONN representation given by

$$\begin{aligned}
\dot{\chi}_1 &= \mathbf{f}_1^*(\chi_1, \mathbf{w}^*) + \mathbf{B}_1^*(\chi_1, \mathbf{w}^*)\chi_2 + \nu_1(t) \\
\dot{\chi}_2 &= \mathbf{f}_2^*(\chi_1, \chi_2 \mathbf{w}^*) + \mathbf{B}_2^*(\chi_1, \chi_2, \mathbf{w}^*)\chi_3 + \nu_2(t) \\
&\vdots \\
\dot{\chi}_i &= \mathbf{f}_i^*(\chi_1, \chi_2, ..., \chi_i, \mathbf{w}^*) + \mathbf{B}_i^*(\chi_1, \chi_2, ..., \chi_i, \mathbf{w}^*)\chi_{i+1} + \nu_i(t) \\
&\vdots \\
\dot{\chi}_r &= \mathbf{f}_r^*(\chi, \mathbf{w}^*) + \mathbf{B}_r^*(\chi, \mathbf{w}^*)\mathbf{u} + \nu_r(t)
\end{aligned} \tag{3.5}$$

Let define $q_1 = n_1$, $q_2 = q_1 + n_2$, ..., $q_i = q_{i-1} + n_i$,...,$q_r = n$, with $i = 1, 2, ..., r$. Then $\mathbf{f}_i^* (\boldsymbol{\chi}_1, \boldsymbol{\chi}_2, ..., \boldsymbol{\chi}_i, \mathbf{w}) = \mathbf{A}_i \boldsymbol{\chi}_i + [\lambda_{(q_i+1)}(\mathbf{w}^*_{(q_i+1)}, \boldsymbol{\chi}_1, \boldsymbol{\chi}_2, ..., \boldsymbol{\chi}_i), \lambda_{(q_i+2)}(\mathbf{w}^*_{(q_i+2)}, \boldsymbol{\chi}_1, \boldsymbol{\chi}_2, ..., \boldsymbol{\chi}_i),..., \lambda_{(q_i+n_i)}(\mathbf{w}^*_{(q_i+n_i)}, \boldsymbol{\chi}_1, \boldsymbol{\chi}_2, ..., \boldsymbol{\chi}_i)]^\top$, $\mathbf{A}_i$=$diag\{-a_{(q_i+1)}, -a_{(q_i+2)}, ..., -a_{(q_i+n_i)}\}$ $i = 1, 2, ..., r$. Where, $\lambda_i(\mathbf{w}_i^*, \boldsymbol{\chi})$ $i = 1, 2, ..., n$ and the entries of $\mathbf{B}_i^* (\boldsymbol{\chi}_1, \boldsymbol{\chi}_2, ...\boldsymbol{\chi}_i, \mathbf{w}^*) \boldsymbol{\chi}_{i+1}$ $i = 1, 2, ..., r$ are of high-order connections, with $\mathbf{w}_1^*$, $\mathbf{w}_2^*$,...,$\mathbf{w}_n^*$ as the optimal weight vectors and $\boldsymbol{\nu}_i(t)$ is the vector of modeling error terms; Additionally we assume that $||\boldsymbol{\nu}_i(t)|| \leq d_{0i}$ for all time.

So we propose the RHONN identifier with the Block Controllable form, expressed by

$$\begin{aligned}
\dot{\mathbf{x}}_1 &= \tilde{\mathbf{f}}_1 (\mathbf{x}_1, \boldsymbol{\chi}_1, \mathbf{w}) + \tilde{\mathbf{B}}_1 (\boldsymbol{\chi}_1, \mathbf{w}) \boldsymbol{\chi}_2 \\
\dot{\mathbf{x}}_2 &= \tilde{\mathbf{f}}_2 (\mathbf{x}_2, \boldsymbol{\chi}_1, \boldsymbol{\chi}_2, \mathbf{w}) + \tilde{\mathbf{B}}_1 (\boldsymbol{\chi}_1, \boldsymbol{\chi}_2, \mathbf{w}) \boldsymbol{\chi}_3 \\
&\vdots \\
\dot{\mathbf{x}}_i &= \tilde{\mathbf{f}}_i (\mathbf{x}_i, \boldsymbol{\chi}_1, \boldsymbol{\chi}_2, ..., \boldsymbol{\chi}_i, \mathbf{w}) + \tilde{\mathbf{B}}_i (\boldsymbol{\chi}_1, \boldsymbol{\chi}_2, ..., \boldsymbol{\chi}_i, \mathbf{w}) \boldsymbol{\chi}_{i+1} \\
&\vdots \\
\dot{\mathbf{x}}_r &= \tilde{\mathbf{f}}_r (\mathbf{x}_r, \boldsymbol{\chi}, \mathbf{w}) + \tilde{\mathbf{B}}_r (\boldsymbol{\chi}, \mathbf{w}) \mathbf{u}
\end{aligned} \tag{3.6}$$

whit $\tilde{\mathbf{f}}_i (\boldsymbol{\chi}_1, \boldsymbol{\chi}_2, ..., \boldsymbol{\chi}_i, \mathbf{w}) = \mathbf{A}_i \mathbf{x}_i + [\lambda_{(q_i+1)} (\mathbf{w}_{(q_i+1)}, \boldsymbol{\chi}_1), \lambda_{(q_i+2)}(\mathbf{w}_{(q_i+2)}, \boldsymbol{\chi}_1), ..., \lambda_{(q_i+n_i)}(\mathbf{w}_{(q_i+n_i)}, \boldsymbol{\chi}_1)]$ $\mathbf{A}_i$=$diag\{-a_{(q_i+1)}, -a_{(q_i+2)}, ... - a_{(q_i+n_i)}\}$ $i = 1, 2, ..., r$. Where, $\lambda_i(\mathbf{w}_i, \boldsymbol{\chi})$ $i = 1, 2, ..., n$ and the entries of $\tilde{\mathbf{B}}_i (\boldsymbol{\chi}_1, \boldsymbol{\chi}_2, ...\boldsymbol{\chi}_i, \mathbf{w}) \boldsymbol{\chi}_{i+1}$ $i = 1, 2, ..., r$ are of high-order connections, and $\mathbf{w}_1$, $\mathbf{w}_2$, ...,$\mathbf{w}_{n_1}$ are the adaptive weight vectors. Additionally, $\tilde{\mathbf{B}}_i (\boldsymbol{\chi}_1, \boldsymbol{\chi}_2, ...\boldsymbol{\chi}_i, \mathbf{w}) \boldsymbol{\chi}_{i+1}$ are full rank matrices and their weights are constants.

As mentioned before, we have separated the tracking problem in two almost independent problems; the first is treated as an identification problem in Chapter 2 and the other is treated as a control problem. In this Section a control technique is developed to drive the neural output $\mathbf{y}_N$, whose dynamics are given by equation (3.5), to a desired output $\mathbf{y}_r$.

3.2.1 Sliding Manifold Design

Define the error vector $\mathbf{z}_1$ as

$$\mathbf{z}_1 = \mathbf{x}_1 - \mathbf{y}_r$$

where $\mathbf{y}_r$ is the desired output. Then the dynamics for $\mathbf{z}_1$ is

$$\dot{\mathbf{z}}_1 = \tilde{\mathbf{f}}_1 (\mathbf{x}_1, \boldsymbol{\chi}_1, \mathbf{w}) + \tilde{\mathbf{B}}_1 (\boldsymbol{\chi}_1, \mathbf{w}) \boldsymbol{\chi}_2 - \dot{\mathbf{y}}_r. \tag{3.7}$$

Substitution of $\mathbf{e}_2 = \mathbf{x}_2 - \boldsymbol{\chi}_2$ in the above equation yields

$$\dot{\mathbf{z}}_1 = \tilde{\mathbf{f}}_1(\mathbf{x}_1, \boldsymbol{\chi}_1, \mathbf{w}) - \dot{\mathbf{y}}_r + \tilde{\mathbf{B}}_1(\boldsymbol{\chi}_1, \mathbf{w})(\mathbf{x}_2 - \mathbf{e}_2). \tag{3.8}$$

Let the fictitious control $\mathbf{x}_2$, in the block (3.7) be chosen as

$$\mathbf{x}_2 = \tilde{\mathbf{B}}_1^{-1}(\boldsymbol{\chi}_1, \mathbf{w})\left[-\tilde{\mathbf{f}}_1(\mathbf{x}_1, \boldsymbol{\chi}_1, \mathbf{w}) + \mathbf{K}_1\mathbf{z}_1 - \tilde{\mathbf{B}}_1(\boldsymbol{\chi}_1, \mathbf{w})\mathbf{e}_2 - \dot{\mathbf{y}}_r\right] + \mathbf{z}_2 \tag{3.9}$$

where $\mathbf{K}_1$ is a Hurwitz matrix, $\tilde{\mathbf{B}}_1^{-1}(\mathbf{x}_1, \mathbf{w})$ is the inverse of $\tilde{\mathbf{B}}_1(\mathbf{x}_1, \mathbf{w})$. We can have a new desired dynamics for the block (3.8) selecting eigenvalues of $\mathbf{K}_1$. Then $\mathbf{z}_1$ dynamics is given by

$$\dot{\mathbf{z}}_1 = \mathbf{K}_1\mathbf{z}_1 + \tilde{\mathbf{B}}_1(\boldsymbol{\chi}_1, \mathbf{w})\mathbf{z}_2.$$

Then $\mathbf{z}_2$ is obtained from (3.9) as

$$\mathbf{z}_2 = \tilde{\mathbf{B}}_1^{-1}(\boldsymbol{\chi}_1, \mathbf{w})\left[\tilde{\mathbf{f}}_1(\mathbf{x}_1, \boldsymbol{\chi}_1, \mathbf{w}) - \mathbf{K}_1\mathbf{z}_1 + \tilde{\mathbf{B}}_1(\boldsymbol{\chi}_1, \mathbf{w})\mathbf{e}_2 + \dot{\mathbf{y}}_r\right] + \mathbf{x}_2 = \boldsymbol{\alpha}_1(\mathbf{x}_1, \boldsymbol{\chi}_1, \mathbf{w}, \dot{\mathbf{y}}_r).$$

Now, differentiating $\mathbf{z}_2$ along the trajectories of (3.6) results

$$\dot{\mathbf{z}}_2 = \bar{\mathbf{f}}_2'(\mathbf{x}_1, \boldsymbol{\chi}, \mathbf{y}_r, \dot{\mathbf{y}}_r, \ddot{\mathbf{y}}_r, \mathbf{w}) + \tilde{\mathbf{B}}_2\boldsymbol{\chi}_3$$

where $\bar{\mathbf{f}}_2'(\mathbf{x}_1, \boldsymbol{\chi}, \mathbf{w}, \mathbf{y}_r, \dot{\mathbf{y}}_r, \ddot{\mathbf{y}}_r) = \dot{\boldsymbol{\chi}}_2 + \left(\frac{\partial \boldsymbol{\alpha}_1}{\partial \mathbf{x}_1}\dot{\mathbf{x}}_1 + \frac{\partial \boldsymbol{\alpha}_1}{\partial \boldsymbol{\chi}_1}\dot{\boldsymbol{\chi}}_1 + \frac{\partial \boldsymbol{\alpha}_1}{\partial \boldsymbol{\chi}_2}\dot{\boldsymbol{\chi}}_2 + \frac{\partial \boldsymbol{\alpha}_1}{\partial \mathbf{w}}\dot{\mathbf{w}} + \frac{\partial \boldsymbol{\alpha}_1}{\partial \mathbf{y}_r}\dot{\mathbf{y}}_r + \frac{\partial \boldsymbol{\alpha}_1}{\partial \dot{\mathbf{y}}_r}\ddot{\mathbf{y}}_r\right)$. Substituting $\mathbf{e}_3 = \mathbf{x}_3 - \boldsymbol{\chi}_3$, the above equation can be rewritten as

$$\begin{aligned} \dot{\mathbf{z}}_2 &= \bar{\mathbf{f}}_2(\mathbf{x}_2, \boldsymbol{\chi}, \mathbf{y}_r, \dot{\mathbf{y}}_r, \ddot{\mathbf{y}}_r, \mathbf{w}) + \tilde{\mathbf{B}}_2(\boldsymbol{\chi}_1, \boldsymbol{\chi}_2, \mathbf{w})(\mathbf{x}_3 - \mathbf{e}_3) \\ \dot{\mathbf{z}}_2 &= \bar{\mathbf{f}}_2^*(\boldsymbol{\chi}, \mathbf{w}^\star, \boldsymbol{\nu}(t)) + \bar{\mathbf{f}}_2'(\mathbf{x}_2, \boldsymbol{\chi}, \mathbf{y}_r, \dot{\mathbf{y}}_r, \ddot{\mathbf{y}}_r, \mathbf{w}) + \tilde{\mathbf{B}}_2(\boldsymbol{\chi}_1, \boldsymbol{\chi}_2, \mathbf{w})\mathbf{x}_3 \end{aligned} \tag{3.10}$$

with $\bar{\mathbf{f}}_2^*(\boldsymbol{\chi}, \mathbf{w}^\star, \boldsymbol{\nu}(t)) = \dot{\boldsymbol{\chi}}_1 + \left(\frac{\partial \boldsymbol{\alpha}_1}{\partial \boldsymbol{\chi}_1}\dot{\boldsymbol{\chi}}_1 + \frac{\partial \boldsymbol{\alpha}_1}{\partial \boldsymbol{\chi}_2}\dot{\boldsymbol{\chi}}_2\right)$ and $\bar{\mathbf{f}}_2(\mathbf{x}^1, \boldsymbol{\chi}, \mathbf{y}_r, \dot{\mathbf{y}}_r, \ddot{\mathbf{y}}_r, \mathbf{w}) = -\mathbf{B}_2(\boldsymbol{\chi}_1, \boldsymbol{\chi}_2, \mathbf{w})\mathbf{e}_3 + \left(\frac{\partial \boldsymbol{\alpha}_1}{\partial \mathbf{x}_1}\dot{\mathbf{x}}_1 + \frac{\partial \boldsymbol{\alpha}_1}{\partial \mathbf{w}}\dot{\mathbf{w}} + \frac{\partial \boldsymbol{\alpha}_1}{\partial \mathbf{y}_r}\dot{\mathbf{y}}_r + \frac{\partial \boldsymbol{\alpha}_1}{\partial \dot{\mathbf{y}}_r}\ddot{\mathbf{y}}_r\right)$ Notice that $\bar{\mathbf{f}}_2$ is measurable and $\bar{\mathbf{f}}_2^*$ is not measurable but it is bounded.

Let the fictitious control $\mathbf{x}_3$, in the block (3.10), be selected as

$$\mathbf{x}_3 = \tilde{\mathbf{B}}_2^{-1}(\boldsymbol{\chi}_1, \mathbf{w})\left[-\bar{\mathbf{f}}_2(\mathbf{x}_2, \boldsymbol{\chi}, \mathbf{y}_r, \dot{\mathbf{y}}_r, \ddot{\mathbf{y}}_r, \mathbf{w}) + \mathbf{K}_2\mathbf{z}_2\right] + \mathbf{z}_3$$

Then the $\mathbf{z}_2$ dynamics is given by

$$\dot{\mathbf{z}}_2 = \mathbf{K}_2\mathbf{z}_2 + \tilde{\mathbf{B}}_2(\boldsymbol{\chi}_1, \boldsymbol{\chi}_2, \mathbf{w})\mathbf{z}_3 + \bar{\mathbf{f}}_2^*(\boldsymbol{\chi}, \mathbf{w}^\star, \boldsymbol{\nu}(t)).$$

We continue along this way until we obtain

$$\dot{\mathbf{z}}_i = \bar{\mathbf{f}}_i(\mathbf{x}_i, \boldsymbol{\chi}, \mathbf{y}_r, \dot{\mathbf{y}}_r, \ddot{\mathbf{y}}_r \ldots \overset{(i)}{\mathbf{y}}_r, \mathbf{w}) + \tilde{\mathbf{B}}_i(\boldsymbol{\chi}_1, \boldsymbol{\chi}_2, \ldots, \boldsymbol{\chi}_i, \mathbf{w})\,\mathbf{x}_{i+1} + \bar{\mathbf{f}}_i^*(\boldsymbol{\chi}, \mathbf{w}^*, \boldsymbol{\nu}(t)) \tag{3.11}$$

where $i = 3, \ldots, r-1$ and $\mathbf{x}_{i+1}$ is the fictitious control for (3.11) giving as

$$\mathbf{x}_{i+1} = \mathbf{B}_i^{-1}(\boldsymbol{\chi}_1, \boldsymbol{\chi}_2, \ldots, \boldsymbol{\chi}_i, \mathbf{w}) \left[-\bar{\mathbf{f}}_i(\mathbf{x}_i, \boldsymbol{\chi}, \mathbf{y}_r, \dot{\mathbf{y}}_r, \ddot{\mathbf{y}}_r \ldots \overset{(i)}{\mathbf{y}}_r, \mathbf{w}) + \mathbf{K}_i \mathbf{z}_i \right] + \mathbf{z}_{i+1}.$$

Then (3.11) can be rewritten as

$$\dot{\mathbf{z}}_i = \mathbf{K}_i \mathbf{z}_i + \mathbf{B}_i(\boldsymbol{\chi}_1, \boldsymbol{\chi}_2, \ldots, \boldsymbol{\chi}_i, \mathbf{w})\,\mathbf{z}_{i+1} + \bar{\mathbf{f}}_i^*(\boldsymbol{\chi}, \mathbf{w}^*, \boldsymbol{\nu}(t)).$$

This procedure yields the nonlinear transformation

$$\begin{aligned}
\mathbf{z}_1 &= \mathbf{x}_1 - \mathbf{y}_r \\
\mathbf{z}_2 &= \mathbf{x}_2 + \tilde{\mathbf{B}}_1^{-1}(\boldsymbol{\chi}_1, \mathbf{w}) \left[\tilde{\mathbf{f}}_1(\mathbf{x}_1, \boldsymbol{\chi}_1, \mathbf{w}) - \mathbf{K}_1 \mathbf{x}_1 \right] \\
\mathbf{z}_3 &= \mathbf{x}_3 + \tilde{\mathbf{B}}_2^{-1}(\boldsymbol{\chi}_1, \boldsymbol{\chi}_2, \mathbf{w}) \left[\bar{\mathbf{f}}_2(\mathbf{x}_2, \boldsymbol{\chi}, \mathbf{y}_r, \dot{\mathbf{y}}_r, \ddot{\mathbf{y}}_r, \mathbf{w}) - \mathbf{K}_2 \mathbf{z}_2 \right] \\
&\vdots \\
\mathbf{z}_i &= \mathbf{x}_i + \tilde{\mathbf{B}}_i^{-1}(\boldsymbol{\chi}_1, \boldsymbol{\chi}_2, \ldots, \boldsymbol{\chi}_i, \mathbf{w}) \left[\bar{\mathbf{f}}_i(\mathbf{x}_i, \boldsymbol{\chi}, \mathbf{y}_r, \dot{\mathbf{y}}_r, \ddot{\mathbf{y}}_r \ldots \overset{(i)}{\mathbf{y}}_r, \mathbf{w}) - \mathbf{K}_i \mathbf{z}_i \right]
\end{aligned} \tag{3.12}$$

where $i = 3, \ldots, r-1$.

The dynamics for system(3.3) in $\mathbf{z}$-coordinates is

$$\begin{aligned}
\dot{\mathbf{z}}_1 &= \mathbf{K}_1 \mathbf{z}_1 + \tilde{\mathbf{B}}_1 \mathbf{z}_2 \\
\dot{\mathbf{z}}_2 &= \mathbf{K}_2 \mathbf{z}_2 + \tilde{\mathbf{B}}_2 \mathbf{z}_3 + \bar{\mathbf{f}}_2^* \\
\dot{\mathbf{z}}_3 &= \mathbf{K}_3 \mathbf{z}_3 + \tilde{\mathbf{B}}_3 \mathbf{z}_4 + \bar{\mathbf{f}}_3^* \\
&\vdots \\
\dot{\mathbf{z}}_{r-1} &= \mathbf{K}_{r-1} \mathbf{z}_{r-1} + \tilde{\mathbf{B}}_{r-1} \mathbf{z}_r + \bar{\mathbf{f}}_i^*
\end{aligned} \tag{3.13}$$

$$\dot{\mathbf{z}}_r = \bar{\mathbf{f}}_r + \tilde{\mathbf{B}}_r \mathbf{u} + \bar{\mathbf{f}}_r^*. \tag{3.14}$$

3.2.2 Discontinuous Feedback

Now, the variable structure control action is selected by

$$\mathbf{u} = -u_0 \tilde{\mathbf{B}}_r^{-1} \mathbf{sign}\left(\mathbf{z}_r\right). \tag{3.15}$$

Substitution of (3.15) into (3.14) yields

$$\dot{\mathbf{z}}_2 = \bar{\mathbf{f}}_r + \bar{\mathbf{f}}_r^* - u_0 \mathbf{sign}(\mathbf{z}_r). \tag{3.16}$$

Assume now that the following bound is satisfied

$$\left\|\bar{\mathbf{f}}_r + \bar{\mathbf{f}}_r^*\right\|_2 \leq q_0, \text{and } q_0 > 0 \quad \forall t$$

then the controller (3.15) with u_0 chosen as

$$u_0 \geq \frac{q_0}{\sqrt{n_r}} + d_0, \quad d_0 > 0$$

guarantees a sliding mode on the surface $\mathbf{z}_r = \mathbf{0}$ in a finite time. Indeed, taking the Lyapunov function candidate $V_r = \frac{1}{2}\mathbf{z}_r{}^\top \mathbf{z}_r$, we have that along the trajectories of (3.14)

$$\begin{aligned} \dot{V}_r &= -u_0 \left\|\mathbf{z}_r\right\|_1 + \mathbf{z}_r{}^\top \left[\bar{\mathbf{f}}_r + \bar{\mathbf{f}}_r^*\right] \\ &\leq -[u_0\sqrt{n_r} - q_0] \left\|\mathbf{z}_r\right\|_2 \leq -\frac{d_0}{\sqrt{n_r}} \left\|\mathbf{z}_r\right\|_2 \leq -\frac{\sqrt{2}d_0}{\sqrt{n_r}} \sqrt{V_r} \end{aligned}$$

which shows that the Lyapunov function vanish in a finite time. This fact implies the existence of the sliding motions on the manifold $\mathbf{z}_r = \mathbf{0}$. This motion is described in new variables $\mathbf{z}$ by the linear system $n - n_r$ order

$$\begin{aligned} \dot{\mathbf{z}}_1 &= \mathbf{K}_1\mathbf{z}_1 + \tilde{\mathbf{B}}_1\mathbf{z}_2 \\ \dot{\mathbf{z}}_2 &= \mathbf{K}_2\mathbf{z}_2 + \tilde{\mathbf{B}}_2\mathbf{z}_3 + \bar{\mathbf{f}}_2^* \\ \dot{\mathbf{z}}_3 &= \mathbf{K}_3\mathbf{z}_3 + \tilde{\mathbf{B}}_3\mathbf{z}_4 + \bar{\mathbf{f}}_3^* \\ &\vdots \\ \dot{\mathbf{z}}_{r-1} &= \mathbf{K}_{r-1}\mathbf{z}_{r-1} + \bar{\mathbf{f}}_{r-1}^*. \end{aligned} \tag{3.17}$$

Now, rewriting the system (3.17) as

$$\begin{bmatrix} \dot{\mathbf{z}}_1 \\ \dot{\mathbf{z}}_2 \\ \vdots \\ \dot{\mathbf{z}}_{r-1} \end{bmatrix} = \begin{bmatrix} \mathbf{K}_1 & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{K}_2 & \cdots & \mathbf{0} \\ \vdots & \vdots & & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{K}_{r-1} \end{bmatrix} \begin{bmatrix} \mathbf{z}_1 \\ \mathbf{z}_2 \\ \vdots \\ \mathbf{z}_{r-1} \end{bmatrix} + \begin{bmatrix} \tilde{\mathbf{B}}_1\mathbf{z}_2 \\ \tilde{\mathbf{B}}_2\mathbf{z}_3 + \bar{\mathbf{f}}_2^* \\ \vdots \\ \bar{\mathbf{f}}_{r-1}^* \end{bmatrix}$$

which can be consider as a linear stable one, perturbed by the last term. So, if such term is bounded, we can conclude that the trajectories of $\left[\mathbf{z}_{r-2}^\top, \mathbf{z}_{r-1}^\top, ..., \mathbf{z}_1^\top\right]^\top$ are ultimate bounded [26].

3.3 Neural Block Control with Relative Degree 2

First, recall that the control problem with relative degree is trivial for VSC. Now, let consider a system with NBC form (3.3) and relative degree equal to 2 and $n_1 = n_2$. Additionally, assume that the first block of this system has a Block Controllable (BC) RHONN representation given by

$$\dot{\chi}_1 = \mathbf{f}_1^*(\chi_1, t) + \mathbf{B}_1^*(\chi_1, t)\chi_2 + \nu(t) \tag{3.18}$$

where $\mathbf{f}_1^*(\chi_1, t) = \mathbf{A}\chi_1 + [\lambda_1(\mathbf{w}_1^*, \chi_1), \lambda_2(\mathbf{w}_2^*, \chi_1), ..., \lambda_{n_1}(\mathbf{w}_{n_1}^*, \chi_1)]^\top$, $\mathbf{A} = diag\{-a_1, -a_2, ... - a_{n_1}\}$, $\lambda_i(\mathbf{w}_i^*, \chi_1)$ $i = 1, 2, ..., n_1$ and the entries of $\mathbf{B}_1^*(\chi_1, t)\chi_2$ are of high order terms. With $\mathbf{w}_1^*, \mathbf{w}_2^*, ..., \mathbf{w}_{n_1}^*$ as the optimal weight vectors and $\nu(t) = [\nu_1(t), \nu_2(t), ..., \nu_{n_1}(t)]^\top$ is the vector of modeling error terms.

So we propose, for the block (3.18), a partial state RHONN identifier with the Block Controllable form, expressed by

$$\dot{\mathbf{x}}_1 = \tilde{\mathbf{f}}_1(\mathbf{x}_1, \chi_1, t) + \tilde{\mathbf{B}}_1(\chi_1, t)\chi_2 \tag{3.19}$$

where $\tilde{\mathbf{f}}_1(\mathbf{x}_1, \chi_1, t) = \mathbf{A}\mathbf{x}_1 + [\lambda_1(\mathbf{w}_1, \chi_1), \lambda_2(\mathbf{w}_2, \chi_1), ..., \lambda_{n_1}(\mathbf{w}_{n_1}, \chi_1)]^\top$ $\lambda_i(\mathbf{w}_i, \chi_1)$ $i = 1, 2, ..., n_1$ and the entries of $\mathbf{B}_1(\chi_1, t)\chi_2$ are the same of high-order terms than in (3.18), but $\mathbf{w}_1$, $\mathbf{w}_2, ..., \mathbf{w}_{n_1}$ are the adaptive weight vectors

3.3.1 Sliding Manifold Design

Define the error vector $\mathbf{z}_1$ as

$$\mathbf{z}_1 = \mathbf{x}_1 - \mathbf{y}_r.$$

Then the dynamics for $\mathbf{z}_1$ is

$$\dot{\mathbf{z}}_1 = \tilde{\mathbf{f}}_1(\mathbf{x}_1, \chi_1, \mathbf{w}) + \tilde{\mathbf{B}}_1(\chi_1, \mathbf{w})\chi_2 - \dot{\mathbf{y}}_r. \tag{3.20}$$

let the fictitious control χ_2, in the block (3.19), be chosen as

$$\chi_2 = \tilde{\mathbf{B}}_1^{-1}(\chi_1, \mathbf{w})\left[-\tilde{\mathbf{f}}_1(\mathbf{x}_1, \chi_1, \mathbf{w}) + \mathbf{K}_1\mathbf{z}_1 - \dot{\mathbf{y}}_r\right] + \mathbf{z}_2 \tag{3.21}$$

where $\mathbf{K}_1$ is a Hurwitz matrix, $\tilde{\mathbf{B}}_1^{-1}(\mathbf{x}_1, \mathbf{w})$ is the inverse of $\tilde{\mathbf{B}}_1(\mathbf{x}_1, \mathbf{w})$. It is worth mentioning that due to the time-varying nature of some RHONN parameters, $\tilde{\mathbf{B}}_1(\mathbf{x}_1, \mathbf{w})$ may lose rank, so

we assume that $\mathbf{w}$ varies in such a way that $\tilde{\mathbf{B}}_1(\mathbf{x}_1, \mathbf{w})$ keeps its rank for all time. With the fictitious control $\boldsymbol{\chi}_2$ (3.21), we have a new desired dynamics for the block (3.20) selecting the eigenvalues of $\mathbf{K}_1$.

Then $\mathbf{z}_2$ is obtained from (3.21) as

$$\mathbf{z}_2 = \tilde{\mathbf{B}}_1^{-1}(\boldsymbol{\chi}_1, \mathbf{w}) \left[\tilde{\mathbf{f}}_1(\mathbf{x}_1, \boldsymbol{\chi}_1, \mathbf{w}) - \mathbf{K}_1 \mathbf{z}_1 + \dot{\mathbf{y}}_r\right] + \boldsymbol{\chi}_2$$

and the dynamics for $\mathbf{z}_1$ is rewritten as

$$\dot{\mathbf{z}}_1 = \mathbf{K}_1 \mathbf{z}_1 + \tilde{\mathbf{B}}_1(\boldsymbol{\chi}_1, \mathbf{w})\, \mathbf{z}_2.$$

Now, differentiating $\mathbf{z}_2$ along the trajectories of (3.3) it results

$$\dot{\mathbf{z}}^2 = \bar{\mathbf{f}}_2(\mathbf{x}^1, \boldsymbol{\chi}, \mathbf{y}_r, \dot{\mathbf{y}}_r, \ddot{\mathbf{y}}_r, \mathbf{w}) + \mathbf{B}_2 \mathbf{u} \tag{3.22}$$

where $\bar{\mathbf{f}}_2(\mathbf{x}_1, \boldsymbol{\chi}, \mathbf{w}, \mathbf{y}_r, \dot{\mathbf{y}}_r, \ddot{\mathbf{y}}_r) = \mathbf{f}_2(\boldsymbol{\chi}) - \left(\frac{\partial \alpha_1}{\partial \mathbf{x}_1}\tilde{\mathbf{f}}_1 + \frac{\partial \alpha_1}{\partial \boldsymbol{\chi}_1}\mathbf{f}_1 + \frac{\partial \alpha_1}{\partial \boldsymbol{\chi}_1}\mathbf{f}_2 + \frac{\partial \alpha_1}{\partial \mathbf{w}}\dot{\mathbf{w}} + \frac{\partial \alpha_1}{\partial \mathbf{y}_r}\dot{\mathbf{y}}_r + \frac{\partial \alpha_1}{\partial \dot{\mathbf{y}}_r}\ddot{\mathbf{y}}_r\right)$

3.3.2 Discontinuous Feedback

Now, the VSC action is selected as

$$\mathbf{u} = -u_0 \mathbf{B}_2^{-1} \mathbf{sign}\left(\mathbf{z}^2\right) \tag{3.23}$$

Substitution of (3.23) into (3.22) yields

$$\dot{\mathbf{z}}_2 = \bar{\mathbf{f}}_2(\mathbf{x}_1, \boldsymbol{\chi}, \mathbf{w}, \mathbf{y}_r, \dot{\mathbf{y}}_r, \ddot{\mathbf{y}}_r) - u_0 \mathbf{sign}(\mathbf{z}_2). \tag{3.24}$$

Let assume now that the following bound is satisfied

$$\left\|\bar{\mathbf{f}}_2(\mathbf{x}_1, \boldsymbol{\chi}, \mathbf{w}, \mathbf{y}_r, \dot{\mathbf{y}}_r, \ddot{\mathbf{y}}_r)\right\|_2 \leq q_0, \text{and } q_0 > 0 \quad \forall t \tag{3.25}$$

then the controller (3.23) with u_0 chosen as

$$u_0 \geq \frac{q_0}{\sqrt{n_r}} + d_0, \quad d_0 > 0 \tag{3.26}$$

guarantees a sliding mode on the surface $\mathbf{z}_2 = \mathbf{0}$ in a finite time. Indeed, taking the Lyapunov function candidate $V_r = \frac{1}{2}\mathbf{z}_2{}^\top \mathbf{z}_2$, we have that along the trajectories of (3.24)

$$\begin{aligned} \dot{V}_r &= -u_0 \left\|\mathbf{z}_2\right\|_1 + \mathbf{z}_2{}^\top \left[\bar{\mathbf{f}}_2(\mathbf{x}_1, \boldsymbol{\chi}, \mathbf{w}, \mathbf{y}_r, \dot{\mathbf{y}}_r, \ddot{\mathbf{y}}_r)\right] \\ &\leq -[u_0\sqrt{n_r} - q_0] \left\|\mathbf{z}_2\right\|_2 \leq -\frac{d_0}{\sqrt{n_r}} \left\|\mathbf{z}_2\right\|_2 \leq -\frac{\sqrt{2} d_0}{\sqrt{n_r}} \sqrt{V_r} \end{aligned}$$

which shows that the Lyapunov function vanishes in a finite time. This fact implies the existence of the sliding motions on the surface $\mathbf{z}_2 = \mathbf{0}$. This motion is described in new variables $\mathbf{z}_1$ by the linear system n_1-*th* order

$$\dot{\mathbf{z}}_1 = \mathbf{K}_1 \mathbf{z}_1 \tag{3.27}$$

Therefore, we have guaranteed the convergence of the neural output $\mathbf{y}_N$ to the desired output $\mathbf{y}_r$. Note that if the relative degree is 2 then the disturbances $\left[\bar{\mathbf{f}}_2^{*\top}, ..., \bar{\mathbf{f}}_{r-1}^{*\top}\right]^\top$ of the general NBC, do not appear in the dynamics of $\mathbf{z}_1$, assuring its convergence to zero.

3.3.3 Stability Analysis without Modelling Error

Lemma 3.1 *Let a system with BCF with output* $\mathbf{y}_N = \boldsymbol{\chi}_1$, *relative degree equal to 2 and* $n_1 = n_2 = m$. *Additionally, its first block has a neural representation as (3.18) and the modelling error terms are zeros. Suppose that*

- *A partial state RHONN as described in (3.19) is used to identify (3.18), the RHONN adaptive parameters are updated as in (2.13).*
- *The desired output* $\mathbf{y}_r$, $\dot{\mathbf{y}}_r$ *and* $\ddot{\mathbf{y}}_r$ *are bounded.*
- *The matrix* $\tilde{\mathbf{B}}_1(\boldsymbol{\chi}_1, t)$ *does not lose rank, for all* t.
- *The stability margin (3.26) is satisfied.*
- *A neural block controller based on model (3.19) is applied to the system, as done in Section 3.3.*

Then, the output $\mathbf{y}_N$ *tracks for a desired output* $\mathbf{y}_r$.

Proof. Let propose the following like Lyapunov candidate function.

$$V = \frac{1}{2}\mathbf{e}^\top \mathbf{e} + \frac{1}{2}\sum_{i=1}^{n_1} \tilde{\mathbf{w}}_i^\top \Gamma_i \tilde{\mathbf{w}}_i + \frac{1}{2}\mathbf{z}_1^\top \mathbf{P} \mathbf{z}_1 \tag{3.28}$$

where $\mathbf{R}$ is a symmetric positive definite matrix and $\mathbf{e} = [e_1, e_2, .., e_{n_1}]^\top$

Assuming that all of the modelling error terms $\boldsymbol{\nu}(t)$ are zeros and the manifold $\mathbf{z}_2 = 0$ has been reached, the time derivative of (3.28) along the trajectories of (2.9), (2.13) and (3.27), is expressed as

$$\dot{V} = -\mathbf{e}^\top \mathbf{Q} \mathbf{e} - \mathbf{z}_1 \mathbf{S} \mathbf{z}_1$$

where $\mathbf{Q} = -(\mathbf{A} + \mathbf{A}^\top)$, and $\mathbf{S} = -(\mathbf{RK} + \mathbf{K}^\top\mathbf{R})$ are a positive definite matrices and $\mathbf{A} = diag\{a_1, a_2, \ldots, a_{n_1}\}$.

Then, by using Lemma 2.1, we conclude that $\mathbf{e}$ and $\mathbf{z}_1$ are asymptotically stable in the manifold $\mathbf{z}_2 = 0$. Therefore $\mathbf{y}_N \rightarrow \mathbf{y}_P$ and $\mathbf{y}_N \rightarrow \mathbf{y}_r$, so we conclude that $\mathbf{y}_P \rightarrow \mathbf{y}_r$. ■

3.3.4 Stability Analysis with Modelling Error

Lemma 3.2 *Let assume the same BC system and the same assumptions as in Lemma 3.1, with the parameters update law 2.14 and the modeling error terms not zero but bounded. Then the vector* $[\mathbf{e}^\top, \mathbf{w}^\top, \mathbf{w}_2^\top, \ldots, \mathbf{w}_{n_1}^\top, \mathbf{z}_1^\top]^\top$ *converges to a bounded set*

Proof. Recall Lemma 2.2, then $[e_i\ \tilde{\mathbf{w}}_i^\top]^\top$ $i = 1, 2, \ldots, n_1$ converges into the residual set

$$D_i = \left\{ [e_i\ \tilde{\mathbf{w}}_i^\top]^\top : V_i \leq \frac{1}{\alpha_i} \left(\frac{1}{2}\sigma_{io}||\mathbf{w}_i^*||^2 + \frac{d_0^2}{2} \right) \right\}$$

Assuming that the matrix $\tilde{\mathbf{B}}_1(\boldsymbol{\chi}_1, t)$ has full rank and the stability margin (3.26) is satisfied, the neural block controller of this Section guarantees the convergence of $\mathbf{z}_1$ to zero. Hence the proof is complete. ■

Chapter 4

Induction Motor Neural Control

Induction motors are widely used in industrial applications due to their reliability, simpler construction and reduced cost with respect, for example, to D.C. motors. However, the control of induction motors can be a difficult problem since the dynamics are highly nonlinear and the parameters, mainly rotor resistance and load torque, could de considered as time varying. Provided that all state variables are measured and all parameters are known, different controllers can be, and indeed have been proposed, including the field oriented controller [3], [32], VSC sliding mode controller [56], [59], and more recently, exact input-output linearization [31], [37] and passivity-based controllers [43].

Comparing with other approaches, VSC sliding controller achieves robust performance properties for the closed-loop system. However, this robustness property can be provided only with respect to uncertainties which satisfy the so-called matching condition. On the other hand, the uncertainty, caused by rotor resistance and load torque variations, does not satisfy the matching condition. On-line identification of the load torque was proposed first in [56]. In [40], a nonlinear input-output state feedback linearizing control scheme, which is adaptive with respect to both load torque and rotor resistance, was proposed. In our researches, we assume that all of the induction motor parameters can change in a wide range.

In the Section 4.4 of this chapter the Neural Block Control [52] [51] approach is used to design a nonlinear sliding surface such that the resulting sliding mode dynamics is described by a desired linear system. An alternative control law is derived in Section 4.5, using Singular Sliding Modes technique [11] [35].

For some applications, it is important however to take into account the discrete feature of the electrical drives, as done in [9], where the Direct Torque Control (DTC), a heuristic control strategy for high power induction motor application, is proposed. The DTC is particularly appealing for slow-sampling applications, where the average approximation used for the im-

plementation of modulation-based control may be inadequate. There has been some efforts to study the DTC controller stability, for example [44] and [6]. As well as DTC strategy, the approach of Section 4.6 switches the power electronic devices just one time, per sample time.

The proposed neural identifier and control strategies allow trajectory tracking for induction motors.

4.1 Induction Motor Model

For electrical motors applications, the drives are basically constant voltage sources connected to the motor windings by power electronic switching devices (BJT, GTO, IGBT, etc.). Figure 4.1 shows a switched inverter, connected to a three-phase induction motor [33]. The switching elements may be, for example, IGBT(Insulated-Gate Bipolar Transistor). Each IGBT pair can be manipulated by one of the control binary variables a, b and c. The power transistors are commuted from the ON (saturation) to the OFF(cut-off) state and vice versa, depending on theirs corresponding binary variables, as illustrated in Figure 4.1. All binary variables may change their states independently. Hence, there are eight possible combinations.

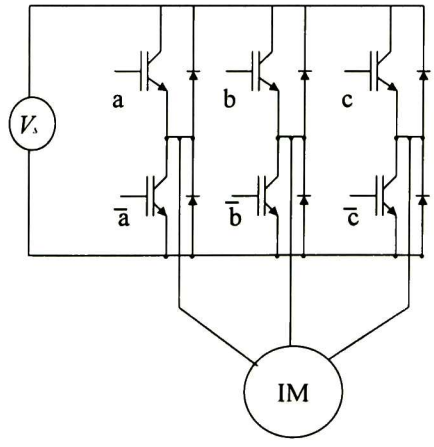


Figure 4.1: The inverter and induction motor connection.

A three-phase circuit can be reduced to a more convenient two-phase model. Let define the input voltage vector $\mathbf{u} = [u_\alpha \; u_\beta]^\top$, in the two-phase $\alpha - \beta$ reference frame [5], where u_α and u_β stand, respectively, for the voltage applied to the induction motor stator windings. Then, the available input vectors are restricted to a discrete set U. The relation between these voltages vectors and the binary variables (a, b and c) is formulated as

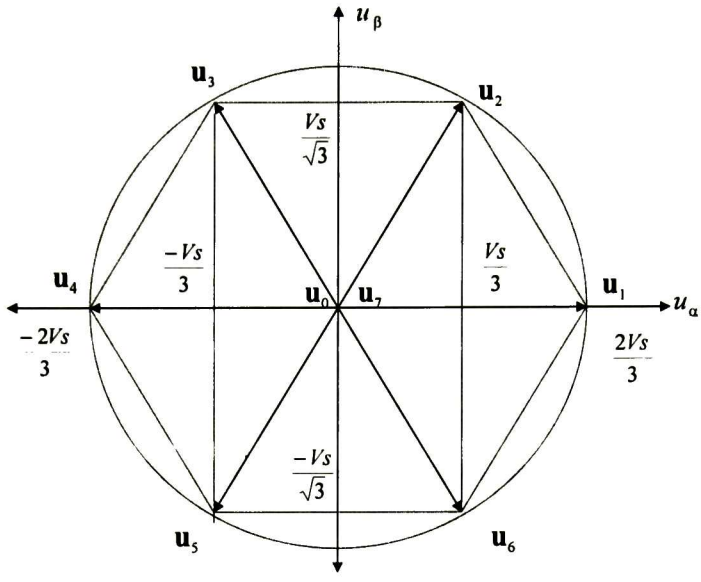


Figure 4.2: Avaliable Stator Voltages Vectors

abc	U	u_α	u_β
000	$\mathbf{u}_0$	0	0
100	$\mathbf{u}_1$	$2V_s/3$	0
110	$\mathbf{u}_2$	$V_s/3$	$V_s/\sqrt{3}$
010	$\mathbf{u}_3$	$-V_s/3$	$V_s/\sqrt{3}$
011	$\mathbf{u}_4$	$-2V_s/3$	0
001	$\mathbf{u}_5$	$-V_s/3$	$-V_s/\sqrt{3}$
101	$\mathbf{u}_6$	$V_s/3$	$-V_s/\sqrt{3}$
111	$\mathbf{u}_7$	0	0

Table 1. Binary variables and theirs corresponding input voltages vectors

where V_s is given by the constant voltage source, which feeds the inverter. Figure 4.2 is the phase portrait of the available input vectors.

In many applications the discrete constrains (Table 1) imposed by the inverter (Figure 4.1) are solved by using a PWM (Pulse Width Modulator) approach [33]. So one may assume that the input $\mathbf{u}$ can be any bounded time function, as done in Section 4.4 and 4.5. In Section 4.6 the control law is derived considering such constrains without using PWM.

The following equation set is presented in the stator-fixed $\alpha-\beta$ coordinate system (see for

instance [5]); they describes the induction motor dynamics

$$\begin{aligned}
\frac{d\omega}{dt} &= c_1(\psi_\alpha i_\beta - \psi_\beta i_\alpha) - c_0 T_L \\
\frac{d\psi_\alpha}{dt} &= -c_2\psi_\alpha - n_p\omega\psi_\beta + c_3 i_\alpha \\
\frac{d\psi_\beta}{dt} &= -c_2\psi_\beta + n_p\omega\psi_\alpha + c_3 i_\beta \\
\frac{di_\alpha}{dt} &= c_4\psi_\alpha + c_5 n_p\omega\psi_\beta - c_5 i_\alpha + c_7 u_\alpha \\
\frac{di_\beta}{dt} &= c_4\psi_\beta - c_5 n_p\omega\psi_\alpha - c_5 i_\beta + c_7 u_\beta
\end{aligned} \tag{4.1}$$

where ω represents the angular velocity of the motor shaft, ψ_α and ψ_β are, respectively, the rotor magnetic flux leakage components, i_α and i_β are, respectively, the stator current components, u_α and u_β stand, respectively, for the voltage applied on the stator windings, and T_L represents the load torque perturbation. The constants c_i, $i = 0, ..., 7$ are defined as follows:

$$\begin{aligned}
c_0 &= \frac{1}{J}, c_1 = \frac{3}{2}\frac{Mn_p}{JL_r}, c_2 = \frac{R_r}{L_r}, c_3 = \frac{R_r M}{L_r}, c_4 = \frac{R_r}{L_r}\frac{M}{L_s L_r - M^2} \\
c_5 &= \frac{M}{L_s L_r - M^2}, c_6 = \frac{R_s L_r^2 + R_r M^2}{L_s(L_s L_r - M^2)}, c_7 = \frac{L_r}{L_s L_r - M^2}
\end{aligned}$$

where L_s, L_r and M, are respectively the stator and rotor inductances and mutual inductance between the rotor and the stator. R_s and R_r, the stator and rotor resistances, J the rotor moment of inertia and n_p the number of stator winding pole pairs.

It is more suitable for neural network identification to present the induction motor model in new variables defined as $\chi_1 = \omega$, $\chi_2 = \psi_\alpha$, $\chi_3 = \psi_\beta$, $\chi_4 = i_\alpha$, $\chi_5 = i_\beta$. Henceforth, the model (4.1) can be rewritten as

$$\begin{aligned}
\dot{\chi}_1 &= c_1(\chi_2\chi_5 - \chi_3\chi_4) - c_0 T_L \\
\dot{\chi}_2 &= -c_2\chi_2 - n_p\chi_1\chi_3 + c_3\chi_4 \\
\dot{\chi}_3 &= -c_2\chi_3 + n_p\chi_1\chi_2 + c_3\chi_5 \\
\dot{\chi}_5 &= c_4\chi_2 + c_5 n_p\chi_1\chi_3 - c_6\chi_4 + c_7 u_\alpha \\
\dot{\chi}_6 &= c_4\chi_3 - c_5 n_p\chi_1\chi_2 - c_6\chi_5 + c_7 u_\beta
\end{aligned} \tag{4.2}$$

This system is in a quasi-nonlinear block controllable (NBC) form. Then the above model can be expressed in block form as

$$\dot{\boldsymbol{\chi}}^1 = \mathbf{f}_1(\boldsymbol{\chi}^1) + \mathbf{B}_1(\boldsymbol{\chi}^1)\boldsymbol{\chi}^2 \tag{4.3}$$

$$\dot{\boldsymbol{\chi}}^2 = \mathbf{f}_2(\boldsymbol{\chi}^1, \boldsymbol{\chi}^2) + \mathbf{B}_2\mathbf{u} \tag{4.4}$$

with $\boldsymbol{\chi} = [\boldsymbol{\chi}_1^\top \ \boldsymbol{\chi}_2^\top]^\top$ $\boldsymbol{\chi}_1 = [\chi_1 \ \chi_2 \ \chi_3]^\top$, $\boldsymbol{\chi}_1 = [\chi_4 \ \chi_5]^\top$ $\mathbf{u} = [u_\alpha \ u_\beta]^\top$ $n_1 = 3$ and $n_2 = m = 2$.

$$\mathbf{f}_1(\boldsymbol{\chi}_1) = \begin{bmatrix} -c_0 T_L \\ -c_2\chi_2 - n_p\chi_1\chi_3 + c_3\chi_4 \\ -c_2\chi_3 + n_p\chi_1\chi_2 + c_3\chi_5 \end{bmatrix} \quad \mathbf{B}_1(\boldsymbol{\chi}_1) = \begin{bmatrix} -c_1\chi_3 & c_1\chi_2 \\ c_3 & 0 \\ 0 & c_3 \end{bmatrix}$$

$$\mathbf{f}_2(\boldsymbol{\chi}_1, \boldsymbol{\chi}_1) = \begin{bmatrix} c_4\chi_2 + c_5 n_p\chi_1\chi_3 - c_6\chi_4 \\ c_4\chi_3 - c_5 n_p\chi_1\chi_2 - c_6\chi_5 \end{bmatrix} \quad \mathbf{B}_2 = \begin{bmatrix} c_7 & 0 \\ 0 & c_7 \end{bmatrix}$$

Commonly, induction motor applications require not only shaft speed regulation, but also flux magnitude $\phi = \chi_2^2 + \chi_3^2$ regulation. Based on this model, the so-called dynamic block controllable neural network is proposed in the Section 4.3.

4.2 Flux Observer

Since the currents and velocity are the only measurable variables the rotor fluxes estimation is required for neural networks identification. This flux estimator was proposed in [36]; it is a partial state observer with adjustable convergence rate. This features enables to reduce the number of calculations comparing with a full state observer. To obtain the flux estimation, only the stator currents dynamics is used. The proposed observer has the following form

$$\begin{aligned} \dot{\tilde{\chi}}_4 &= -c_5\chi_4 + c_6 u_1 + v_\alpha \\ \dot{\tilde{\chi}}_5 &= -c_5\chi_5 + c_6 u_2 + v_\beta \end{aligned}$$

where $\tilde{\chi}_4$ and $\tilde{\chi}_5$ are the estimation of currents χ_4 and χ_5 and $\mathbf{v} = [v_\alpha \ v_\beta]^\top$is the observer input.

Let define current observer error as $\varepsilon_\alpha = \chi_4 - \tilde{\chi}_4$ and $\varepsilon_\beta = \chi_5 - \tilde{\chi}_5$, whose dynamics is given by

$$\begin{aligned} \dot{\varepsilon}_\alpha &= c_4\chi_2 + c_5 n_p\chi_1\chi_3 - v_\alpha \\ \dot{\varepsilon}_\alpha &= c_4\chi_3 - c_5 n_p\chi_1\chi_2 - v_\beta. \end{aligned}$$

Then, on the sliding surface ε_α, $\varepsilon_\beta = 0$, the following invariance equation is satisfied

$$\begin{aligned} 0 &= c_4\chi_2 + c_5 n_p\chi_1\chi_3 - v_{\alpha eq} \\ 0 &= c_4\chi_3 - c_5 n_p\chi_1\chi_2 - v_{\beta eq} \end{aligned} \tag{4.5}$$

with $\mathbf{v}_{eq} = [v_{\alpha eq}, \ v_{\beta eq}]^\top$ as the equivalent value of $\mathbf{v}$.

Now, based on unit control, v_α and v_β are selected as

$$v_\alpha = l_1 \frac{\varepsilon_\alpha}{|\varepsilon_\alpha| + \delta} \text{ and } v_\beta = l_2 \frac{\varepsilon_\beta}{|\varepsilon_\beta| + \delta}$$

with l_1, l_2 and δ are positive observer parameters.

If l_1, l_2 are enough large and δ is sufficiently small we guarantee a sliding motion on surface ε_α, $\varepsilon_\beta = 0$. So $\mathbf{v}$ is taken as an estimated of $\mathbf{v}_{eq}$. Therefore we can express (4.5) as

$$\begin{bmatrix} c_4 & c_5 n_p \chi_1 \\ -c_5 n_p \chi_1 & c_4 \end{bmatrix} \begin{bmatrix} \chi_2 \\ \chi_3 \end{bmatrix} = \begin{bmatrix} v_\alpha \\ v_\beta \end{bmatrix}$$

from the this equation, it is possible to obtain the estimation of χ_2 and χ_3 as

$$\begin{bmatrix} \hat{\chi}_2 \\ \hat{\chi}_3 \end{bmatrix} = \frac{1}{c_4^2 + (c_5 n_p \chi_1)^2} \begin{bmatrix} c_4 & -c_5 n_p \chi_1 \\ c_5 n_p \chi_1 & c_4 \end{bmatrix} \begin{bmatrix} v_\alpha \\ v_\beta \end{bmatrix}$$

where the estimated fluxes are $\tilde{\chi}_2$ and $\tilde{\chi}_3$. For the rest of the calculations on this chapter, the estimated fluxes are considered as the real ones.

4.3 Neural Model for Induction Motors

Usually, for nonlinear control systems, the plant model is obtained from the plant physics. For neural control, we propose to build a neural model based on a given plant model structure. The RHONN scheme is very flexible and allows to incorporate a priori information about the plant structure to the neural identifier.

So, let assume that the model (4.2) has the RHONN representation given by

$$\begin{aligned} \dot{\chi}_1 &= -a_1\chi_1 + w_{11}^* S(\chi_1) + w_{12}^* S(\chi_3)\chi_4 + w_{13}^* S(\chi_2)\chi_5 + w_{14}^* + \nu_1(t) \\ \dot{\chi}_2 &= -a_2\chi_2 + w_{21}^* S(\chi_2) + w_{22}^* S(\chi_1)S(\chi_3) + w_{23}^* \chi_4 + \nu_2(t) \\ \dot{\chi}_3 &= -a_3\chi_3 + w_{31}^* S(\chi_3) + w_{32}^* S(\chi_1)S(\chi_2) + w_{33}^* \chi_5 + \nu_3(t) \end{aligned}$$

where $\mathbf{w}_1^* = [w_{11}^*, w_{12}^*, w_{13}^*, w_{14}^*]^\top$ $\mathbf{w}_2^* = [w_{21}^*, w_{22}^*, w_{23}^*]^\top$ and $\mathbf{w}_3^* = [w_{31}^*, w_{32}^*, w_{33}^*]^\top$ are the optimal weight vectors, which are constant and unknown, and $\boldsymbol{\rho}_1 = [S(\chi_1), S(\chi_3)\chi_4, S(\chi_2)\chi_5, 1]^\top$ $\boldsymbol{\rho}_2 = [S(\chi_2), S(\chi_1)S(\chi_3), \chi_4]^\top$ and $\boldsymbol{\rho}_3 = [S(\chi_3), S(\chi_1)S(\chi_2), \chi_5]^\top$ are the high order term vectors. Finally, $\nu_1(t)$, $\nu_2(t)$, $\nu_3(t)$ are the modelling error terms.

Hence, second-order RHONN is used as the identifier. In order to introduce as much information as possible about the induction motor and based on the mathematical model (4.2), the following neural model is proposed

$$\begin{aligned} \dot{x}_1 &= -a_1 x_1 + w_{11} S(\chi_1) + w_{12} S(\chi_3)\chi_4 + w_{13} S(\chi_2)\chi_5 + w_{14} \\ \dot{x}_2 &= -a_2 x_2 + w_{21} S(\chi_2) + w_{22} S(\chi_1)S(\chi_3) + w_{23}\chi_4 \\ \dot{x}_3 &= -a_3 x_3 + w_{31} S(\chi_3) + w_{32} S(\chi_1)S(\chi_2) + w_{33}\chi_5. \end{aligned} \tag{4.6}$$

According to Chapter 2, for this model $\mathbf{w}_1 = [w_{11}, w_{12}, w_{13}, w_{14}]^\top$ $\mathbf{w}_2 = [w_{21}, w_{22}, w_{23}]^\top$ and $\mathbf{w}_3 = [w_{31}, w_{32}, w_{33}]^\top$ are the adaptive RHONN parameters.

For this neural model, x_1 is the neural speed or velocity, and x_2 and x_3 are the neural fluxes, so they are used to identify χ_1, χ_2 and χ_3 respectively. The neural weights $\mathbf{w}_i$ $i = \{1, 2, 3\}$ are updated according to the adaptive law (2.14) from Chapter 3. The term w_{14} is a kind of bias [19] and has the function of compensating the load torque effect; this term does not contradict any assumption used to develop the weight adaptive law.

The output variables to be controlled are the speed χ_1 and the neural flux magnitude ϕ, respectively. Now, let define the neural flux magnitude as $\varphi = x_2^2 + x_3^2$, then, according to Chapter 2, the plant output is $\mathbf{y}_P = [\chi_1 \ \phi]^\top$ the neural output is $\mathbf{y}_N = [x_1 \ \varphi]^\top$ and the reference signal is $\mathbf{y}_r = [\omega_r \ \varphi_r]^\top$

As can be seen in Section 4.4, neural currents are no required, due to the control strategy used in this work.

4.4 Neural Block Control

In this Section, based on the neural identifier (4.6), a control law is developed. Such control law is derived using block control strategy [34] and sliding modes control [59]. Due to the relative degree of the induction motor, the neural block controller for systems with relative degree 2 is applied from Section 3.3.

The neural model (4.6) and the stator currents model (4.4) are combined to obtain a quasi NBC form, consisting of two blocks:

$$\begin{aligned} \dot{\mathbf{x}}_1 &= \tilde{\mathbf{f}}_1(\mathbf{x}_1, \boldsymbol{\chi}_1, \mathbf{w}) + \tilde{\mathbf{B}}_1(\boldsymbol{\chi}_1, \mathbf{w})\boldsymbol{\chi}_2 \\ \dot{\boldsymbol{\chi}}_2 &= \mathbf{f}_2(\boldsymbol{\chi}_1, \boldsymbol{\chi}_2) + \mathbf{B}_2\mathbf{u} \end{aligned} \tag{4.7}$$

with $\mathbf{x}^1 = [x_1, x_2, x_3]^\top$ $\boldsymbol{\chi}_2 = [\chi_4, \chi_5]^\top$ $\mathbf{u} = [u_\alpha, u_\beta]^\top$ $n_1 = 3$ and $n_2 = m = 2$.

$$\tilde{\mathbf{f}}_1(\mathbf{x}_1, \boldsymbol{\chi}_1, \mathbf{w}) = \begin{bmatrix} \tilde{f}_{11}(\mathbf{x}_1, \boldsymbol{\chi}_1, \mathbf{w}) \\ \tilde{f}_{12}(\mathbf{x}_1, \boldsymbol{\chi}_1, \mathbf{w}) \\ \tilde{f}_{13}(\mathbf{x}_1, \boldsymbol{\chi}_1, \mathbf{w}) \end{bmatrix} = \begin{bmatrix} -a_1 x_1 + w_{11} S(\chi_1) + w_{14} \\ -a_2 x_2 + w_{21} S(\chi_2) + w_{22} S(\chi_1) S(\chi_3) \\ -a_3 x_3 + w_{31} S(\chi_3) + w_{32} S(\chi_1) S(\chi_2) \end{bmatrix}$$

$$\tilde{\mathbf{B}}_1(\mathbf{x}_1, \mathbf{w}) = \begin{bmatrix} -w_{12} S(\chi_3) & w_{13} S(\chi_2) \\ w_{23} & 0 \\ 0 & w_{33} \end{bmatrix}$$

For shorter notation all the weights are ordered in one vector $\mathbf{w} = [\mathbf{w}_1^\top \ \mathbf{w}_2^\top \ \mathbf{w}_3^\top]^\top$ This model can be reduced to the NBC-form [34], and therefore the block control methodology is applied. At first, the tracking error for the neural output is rewritten as

$$\mathbf{z}_1 = \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} = \begin{bmatrix} x_1 - \omega_r \\ \varphi - \varphi_r \end{bmatrix} \tag{4.8}$$

Then, the tracking error dynamics can be expressed as the first block of the NBC-form:

$$\dot{\mathbf{z}}_1 = \bar{\mathbf{f}}_1(\mathbf{x}_1, \boldsymbol{\chi}_1, \mathbf{w}, \dot{\mathbf{y}}_r) + \bar{\mathbf{B}}_1(\boldsymbol{\chi}_1, \mathbf{w})\boldsymbol{\chi}_2 \tag{4.9}$$

where $\bar{\mathbf{f}}_1(\mathbf{x}_1, \boldsymbol{\chi}_1, \mathbf{w}, \dot{\mathbf{y}}_r) = \begin{bmatrix} \bar{f}_{11}(\mathbf{x}_1, \boldsymbol{\chi}_1, \mathbf{w}, \dot{\mathbf{y}}_r) \\ \bar{f}_{12}(\mathbf{x}_1, \boldsymbol{\chi}_1, \mathbf{w}, \dot{\mathbf{y}}_r) \end{bmatrix}$, $\bar{\mathbf{B}}_1(\boldsymbol{\chi}_1, \mathbf{w}) = \begin{bmatrix} w_{12}S(\chi_3) & w_{13}S(\chi_2) \\ 2w_{23}\chi_2 & 2w_{33}\chi_3 \end{bmatrix}$, with $\bar{f}_{11}(\mathbf{x}_1, \boldsymbol{\chi}_1, \mathbf{w}, \dot{\mathbf{y}}_r) = -a_1 x_1 + w_{11}S(\chi_1) + w_{14} - \dot{\omega}_r$ and

$$\begin{aligned} \bar{f}_{12}(\mathbf{x}_1, \boldsymbol{\chi}_1, \mathbf{w}, \dot{\mathbf{y}}_r) \ = \ & 2x_2\left(-a_2 x_2 + \ w_{21}S(\chi_2) + w_{22}S(x_1)S(\chi_3)\right) + \\ & 2x_3(-a_3 x_3 + w_{31}S(\chi_3) + w_{32}S(\chi_1)S(\chi_2)) - \dot{\varphi}_r. \end{aligned}$$

Due to time varying nature of RHONN weights, we can not guarantee that rank($\bar{\mathbf{B}}_1$)= 2 for all time, so we assume that those parameters do not change their signs, keeping $\bar{\mathbf{B}}_1$ as a full rank matrix.

4.4.1 Control Law

Following the block control strategy, the quasi-control vector $\boldsymbol{\chi}_2$ is selected as

$$\boldsymbol{\chi}_2 = \begin{bmatrix} \chi_4 \\ \chi_5 \end{bmatrix} = \bar{\mathbf{B}}_1^{-1}(\boldsymbol{\chi}_1, \mathbf{w})\left(-\bar{\mathbf{f}}_1(\mathbf{x}_1, \boldsymbol{\chi}_1, \mathbf{w}, \dot{\mathbf{y}}_r) + \mathbf{K}\mathbf{z}_1\right) + \mathbf{z}_2$$

where $\mathbf{K} = \begin{bmatrix} -k_1 & 0 \\ 0 & -k_2 \end{bmatrix}$, $\bar{\mathbf{B}}_1^{-1} = \frac{1}{\delta}\begin{bmatrix} 2w_{33}\chi_3 & -2w_{23}\chi_2 \\ -w_{13}S(\chi_2) & w_{12}S(\chi_3) \end{bmatrix}$ and $\mathbf{z}_2 = \begin{bmatrix} z_4 \\ z_5 \end{bmatrix}$, with $\delta = 2w_{12}w_{33}\chi_3 S(\chi_3) - 2w_{13}w_{23}\chi_2 S(\chi_2)$, and $k_1, k_2 > 0$.

Then, (4.9) can be rewritten as

$$\dot{\mathbf{z}}_1 = \mathbf{K}\mathbf{z}_1 + \bar{\mathbf{B}}_1(\boldsymbol{\chi}_1, \mathbf{w})\boldsymbol{\chi}_2.$$

Now, the new variables $\mathbf{z}_2$ are expressed as

$$\mathbf{z}_2 = \bar{\mathbf{B}}_1^{-1}(\boldsymbol{\chi}_1, \mathbf{w}) \left(\bar{\mathbf{f}}_1(\mathbf{x}_1, \boldsymbol{\chi}_1, \mathbf{w}, \dot{\mathbf{y}}_r) - \mathbf{K}\mathbf{z}_1\right) + \boldsymbol{\chi}_2 = \boldsymbol{\alpha}_2(\mathbf{x}_1, \boldsymbol{\chi}_1, \boldsymbol{\chi}_2, \mathbf{w}, \mathbf{y}_r, \dot{\mathbf{y}}_r, \boldsymbol{\chi}_2) \tag{4.10}$$

Taking the time derivative of (4.10), the second block of the NBC-form for the variables z_4 and z_5 is presented as

$$\dot{\mathbf{z}}_2 = \bar{\mathbf{f}}_2(\mathbf{x}^1, \boldsymbol{\chi}, \mathbf{r}, \dot{\mathbf{r}}, \ddot{\mathbf{r}}, \mathbf{w}) + \mathbf{B}_2\mathbf{u} \tag{4.11}$$

where $\bar{\mathbf{f}}_2(\mathbf{x}_1, \boldsymbol{\chi}, \mathbf{w}, \mathbf{y}_r, \dot{\mathbf{y}}_r, \ddot{\mathbf{y}}_r) = \mathbf{f}_2(\boldsymbol{\chi}) - \left(\frac{\partial \boldsymbol{\alpha}_1}{\partial \mathbf{x}_1}\bar{\mathbf{f}}_1 + \frac{\partial \boldsymbol{\alpha}_1}{\partial \boldsymbol{\chi}_1}\mathbf{f}_1 + \frac{\partial \boldsymbol{\alpha}_1}{\partial \boldsymbol{\chi}_1}\mathbf{f}_2 + \frac{\partial \boldsymbol{\alpha}_1}{\partial \mathbf{w}}\dot{\mathbf{w}} + \frac{\partial \boldsymbol{\alpha}_1}{\partial \mathbf{y}_r}\dot{\mathbf{y}}_r + \frac{\partial \boldsymbol{\alpha}_1}{\partial \dot{\mathbf{y}}_r}\ddot{\mathbf{y}}_r\right)$

Now, the VSC control strategy formulated as

$$\mathbf{u} = -u_0 \mathbf{sign}\left(\mathbf{z}^2\right)$$

under the condition

$$c_7 u_0 \; > \left|\bar{\mathbf{f}}_2(\mathbf{x}_1, \boldsymbol{\chi}, \mathbf{w}, \mathbf{y}_r, \dot{\mathbf{y}}_r, \ddot{\mathbf{y}}_r)\right|, \quad \forall t$$

guarantees a sliding mode on the surface

$$z_4 = 0, \quad z_5 = 0$$

in finite time. Then the sliding dynamics, for the tracking errors variables z_1 and z_2, is governed by the second order linear system

$$\begin{aligned} \dot{z}_1 &= -k_1 z_1 \\ \dot{z}_2 &= -k_2 z_2 \end{aligned}$$

with desired eigenvalues $-k_1$ and $-k_2$. Hence, we conclude that the neural output tracks the reference, but for the motor outputs we cannot conclude the same. This topic is analyzed in the following Subsection.

4.4.2 Stability Analysis

Assuming that all the modeling error terms are zeros and the surface $\mathbf{z}_1 = 0$ has been reached, let propose the following candidate Lyapunov function.

$$V = \frac{1}{2}\mathbf{e}^\top \mathbf{e} + \frac{1}{2}\sum_{i=1}^{3} \tilde{\mathbf{w}}_i^\top \Gamma_i \tilde{\mathbf{w}}_i + \frac{1}{2}\mathbf{z}_1^\top \mathbf{R}\mathbf{z}_1 \tag{4.12}$$

where $\mathbf{R}$ is a symmetric positive definite matrix and $\mathbf{e} = [e_1, e_2, e_3]^\top$

Under the assumption that all of the modelling error terms are zeros, the time derivative of (4.12) along the trajectories of (2.13), (4.2), (4.6), (4.9) and (4.11), is expressed as

$$\dot{V} = -\mathbf{e}^\top \mathbf{Q}\mathbf{e} - \mathbf{z}_1^\top \mathbf{S}\mathbf{z}_1$$

where $\mathbf{A} = diag\{a_1, a_2, a_3\}$, $\mathbf{Q} = -(\mathbf{A} + \mathbf{A}^\top)$ and $\mathbf{S} = -(\mathbf{R}\mathbf{K} + \mathbf{K}^\top\mathbf{R})$ are positive definite matrices

Then, by using Lemma 3.1, we conclude that $\mathbf{e}$ and $\mathbf{z}_1$ are asymptotically stable in the manifold $\mathbf{z}_1 = 0$. Then $\mathbf{y}_P \to \mathbf{y}_N$ and $\mathbf{y}_N \to \mathbf{y}_r$. So we conclude that $\mathbf{y}_P \to \mathbf{y}_r$.

4.5 Neural Control with Singular Sliding Modes

In this section, the goal is to force the induction motor position to track a specified speed reference without flux tracking. In order to achieve this tracking, a control law based on the singular sliding modes technique [35][54][53] is developed. Given a reference ω_r, we propose the next change of variables

$$z_1 = x_1 - \omega_r$$

Differentiating z_1 along the trajectories of (4.6), we obtain.

$$\dot{z}_1 = -a_1 x_1 + w_{11} S(\chi_1) + w_{12} S(\chi_3)\chi_4 + w_{13} S(\chi_2)\chi_5 - \dot{\omega}_r = -k_1 z_1 + z_2 \tag{4.13}$$

Solving for z_2 the above equation, it yields

$$z_2 = k_1 z_1 - a_1 x_1 + w_{11} S(\chi_1) + w_{12} S(\chi_3)\chi_4 + w_{13} S(\chi_2)\chi_5 - \dot{\omega}_r$$

Then the z_2 dynamics is given by

$$\dot{z}_2 = f_2(x_1, \boldsymbol{\chi}, \mathbf{w}_1, \dot{\omega}_r) + w_{12} c_7 S(\chi_3) u_\alpha + w_{13} c_7 S(\chi_2) u_\beta \tag{4.14}$$

where $f_2(x_1, \boldsymbol{\chi}, \mathbf{w}_1, \ddot{\omega}_r) = \frac{\partial z_2}{\partial x_1}\dot{x}_1 + \frac{\partial z_2}{\partial \boldsymbol{\chi}^1}\left(\mathbf{f}_1(\boldsymbol{\chi}_1) + \mathbf{B}_1(\boldsymbol{\chi}_1)\boldsymbol{\chi}_2\right) + \frac{\partial z_2}{\partial \boldsymbol{\chi}^2}\mathbf{f}_2(\boldsymbol{\chi}) + \frac{\partial z_2}{\partial \mathbf{w}_1}\dot{\mathbf{w}}_1 - \ddot{\omega}_r$

Now, we design the control law

$$\begin{aligned} u_\alpha &= -sign(w_{12} S(\chi_3))\, sign(z_2)\, u_0 \\ u_\beta &= -sign(w_{13} S(\chi_2))\, sign(z_2)\, u_0. \end{aligned}$$

Then (4.14) can be rewritten as

$$\dot{z}_2 = f_2(x_1, \boldsymbol{\chi}, \mathbf{w}_1, \dot{\omega}_r, \ddot{\omega}_r) - u_0 c_7 \left(|w_{12} S(\chi_3)| + |w_{13} S(\chi_2)|\right) sign(z_2)$$

A sufficient condition to guarantee that z_2 converges to zero is

$$(|w_{12} c_7 S(\chi_2)| + |w_{13} c_7 S(\chi_3)|) u_0 > |f_2(x_1, \boldsymbol{\chi}, \mathbf{w}_1, \ddot{\omega}_r)|, \quad \forall t$$

and then (4.13) is expressed by

$$\dot{z}_1 = -k_1 z_1.$$

Therefore, z_1 converges asymptotically to zero.

Because of the induction motor is Bounded-Input-Bounded-Output BIBO [26] and the input is bounded, the internal dynamics are bounded for any time. Notice that this controller does not require the neural fluxes, because the flux magnitude regulation is not one of the goals for this controller.

4.5.1 Stability Analysis

Let propose the following candidate Lyapunov function

$$V = \frac{1}{2} e_1^2 + \frac{1}{2} \tilde{\mathbf{w}}_1^\top \Gamma_1 \tilde{\mathbf{w}}_1 + \frac{1}{2} z_1^2 + \frac{1}{2} z_2^2 \tag{4.15}$$

Assuming that the modeling error terms are zeros and differentiating (4.15) along the trajectories of (2.9), (2.13),(4.2), (4.13) and (4.14). it yields

$$\dot{V} = -a_1 e_1^2 - k_1 z_1^2 + z_2 \left(z_1 + f_2(x_1, \boldsymbol{\chi}, \mathbf{w}_1, \dot{\omega}_r, \ddot{\omega}_r) - u_0 \left(|w_{12} w_{45} S(\chi_3)| + |w_{13} w_{55} S(\chi_2)|\right) sign(z_2)\right)$$

Additionally, assume that the following bounds are satisfied

$$|z_1 + f_2(x_1, \boldsymbol{\chi}, \mathbf{w}_1, \dot{\omega}_r, \ddot{\omega}_r)| \leq q_0, \text{and } q_0 > 0$$

$$|u_0 \left(|w_{12} c_7 S(\chi_3)| + |w_{13} c_7 S(\chi_2)|\right)| \geq q_0 + d_0, \; d_0 > 0, \; i = 1, 2$$

then

$$z_2\left(+z_1 + f_2(x_1, \boldsymbol{\chi}, \mathbf{w}_1, \dot{\omega}_r, \ddot{\omega}_r) - u_0\left(|w_{12}c_7S(\chi_3)| + |w_{13}c_7S(\chi_2)|\right) sign(z_2)\right) \leq -d_0|z_2|$$

which shows that the time derivative $\dot{V}$ satisfies the following inequality

$$\dot{V} \leq -a_1e_1^2 - k_1z_1^2 - d_0|z_2|$$

Analogously to Subsection 4.4, we conclude that, e_1, z_1 and z_2 converge to zero, therefore $\chi_1 \rightarrow \omega_r$.

4.6 Neural Block Control with Discrete Input Constrains

For induction motor drives, the PWM is used to approach the desired control signal. Such modulator forces each power device to switch several times per sample time. However, if the devices are operated at high frequency, this switching PWM strategy may destroy the electronic devices in a short time period, since for every switching (on and off) there is a dissipated power peak and frequently voltage overshooting.

In this Section, based on the Variable Structure Control (VSC) method [58], we propose a discontinuous control strategy for induction motors. Such control scheme enables to reduce the device switching to only one during the sample time, instead of the two switching required by the PWM. This enables to reduce the power electronic devices wearing, and increase theirs life span. This control scheme is based on the Block Control with Input Constrains for induction motors; see Appendix A.

To derive the control law, let start from equation (4.8), but with a slight change, which yields

$$\dot{\mathbf{z}}_1 = \bar{\mathbf{f}}_1(\mathbf{x}_1, \boldsymbol{\chi}_1, \mathbf{w}, \dot{\mathbf{y}}_r) + \bar{\mathbf{B}}_1(\boldsymbol{\chi}_1, \mathbf{w})\boldsymbol{\chi}_1 = -\mathbf{K}\mathbf{z}_1 + \mathbf{z}_2. \tag{4.16}$$

Solving the equation (4.16) for $\mathbf{z}_2$ results

$$\mathbf{z}_2 = \bar{\mathbf{f}}_1(\mathbf{x}_1, \boldsymbol{\chi}_1, \mathbf{w}, \dot{\mathbf{y}}_r) + \bar{\mathbf{B}}_1(\boldsymbol{\chi}_1, \mathbf{w})\boldsymbol{\chi}_2 + \mathbf{K}\mathbf{z}_1.$$

Then the dynamics for $\mathbf{z}_2$ are

$$\dot{\mathbf{z}}_2 = \bar{\mathbf{f}}_2(\mathbf{x}_1, \boldsymbol{\chi}, \mathbf{w}, \mathbf{y}_r, \dot{\mathbf{y}}_r) + \bar{\mathbf{B}}_2(\boldsymbol{\chi}_1, \mathbf{w})\mathbf{u} \tag{4.17}$$

where $\bar{\mathbf{f}}_2(\mathbf{x}_1, \boldsymbol{\chi}, \mathbf{w}, \mathbf{y}_r, \dot{\mathbf{y}}_r) = \begin{bmatrix} \bar{f}_{21}(\mathbf{x}_1, \boldsymbol{\chi}, \mathbf{w}, \mathbf{y}_r, \dot{\mathbf{y}}_r) \\ \bar{f}_{22}(\mathbf{x}_1, \boldsymbol{\chi}, \mathbf{w}, \mathbf{y}_r, \dot{\mathbf{y}}_r) \end{bmatrix} = \frac{\partial \mathbf{z}_2}{\partial \mathbf{x}_1}\tilde{\mathbf{f}}_1 + \frac{\partial \mathbf{z}_2}{\partial \boldsymbol{\chi}_1}\mathbf{f}_1 + \frac{\partial \mathbf{z}_2}{\partial \boldsymbol{\chi}_2}\mathbf{f}_2 + \frac{\partial \mathbf{z}_2}{\partial \mathbf{y}_r}\dot{\mathbf{y}}_r + \frac{\partial \mathbf{z}_2}{\partial \dot{\mathbf{y}}_r}\ddot{\mathbf{y}}_r + \frac{\partial \mathbf{z}_2}{\partial \mathbf{w}}\dot{\mathbf{w}}$ and $\bar{\mathbf{B}}_2(\boldsymbol{\chi}_1, \mathbf{w}) = \bar{\mathbf{B}}_1(\boldsymbol{\chi}_1, \mathbf{w})\mathbf{B}_2$.

Now, we select the desired sliding manifold as $\mathbf{z}_2 = 0$. The next scope is to design a control law which forces the system to reach the desired manifold. This controller must be a logic function which maps from the continuous state $\mathbf{x}$ to the admissible input vectors set U (see Table 1), in order to switch the control binary variables, such that, the desired voltage feeds the induction motor.

4.6.1 Sliding Modes Controller Design

Assumption 4.1. There exists at least one discrete value $\mathbf{u}_k \in U$, such that

$$\mathbf{sign}(\bar{\mathbf{B}}_2(\boldsymbol{\chi}_1, \mathbf{w})\mathbf{u}_k) = -\mathbf{sign}(\mathbf{z}_2), \text{for all } \mathbf{z}_2 \text{ with } \chi_2, \chi_3 \neq 0. \tag{4.18}$$

The controller must select one element of U that satisfies (4.18), and then change the binary variables such that the selected input vector is fed to the stator windings. By using the model (4.2) instead of (4.6), the above assumption is a fact; the proof can be found in Appendix A, Section 2.

Now we analyze the controller stability. Let $\bar{\mathbf{b}}_{21}$, $\bar{\mathbf{b}}_{22}$ be the rows of $\bar{\mathbf{B}}_2(\boldsymbol{\chi}_1, \mathbf{w})$, then it can be expressed as

$$\bar{\mathbf{B}}_2(\boldsymbol{\chi}_1, \mathbf{w})\mathbf{u}_k = \begin{bmatrix} \bar{\mathbf{b}}_{21}\mathbf{u}_k \\ \bar{\mathbf{b}}_{22}\mathbf{u}_k \end{bmatrix} = \begin{bmatrix} sign(\bar{\mathbf{b}}_{21}\mathbf{u}_k)|\bar{\mathbf{b}}_{21}\mathbf{u}_k| \\ sign(\bar{\mathbf{b}}_{22}\mathbf{u}_k)|\bar{\mathbf{b}}_{22}\mathbf{u}_k| \end{bmatrix} = \begin{bmatrix} |\bar{\mathbf{b}}_{21}\mathbf{u}_k| & 0 \\ 0 & |\bar{\mathbf{b}}_{22}\mathbf{u}_k| \end{bmatrix} \begin{bmatrix} sign(\bar{\mathbf{b}}_{21}\mathbf{u}_k) \\ sign(\bar{\mathbf{b}}_{22}\mathbf{u}_k) \end{bmatrix} \tag{4.19}$$

Using (4.18), the expression (4.19) can be formulated as

$$\bar{\mathbf{B}}_2(\boldsymbol{\chi}_1, \mathbf{w})\mathbf{u}_k = \begin{bmatrix} |\bar{\mathbf{b}}_{21}\mathbf{u}_k| & 0 \\ 0 & |\bar{\mathbf{b}}_{22}\mathbf{u}_k| \end{bmatrix} \mathbf{sign}(\bar{\mathbf{B}}_2(\boldsymbol{\chi}_1, \mathbf{w})\mathbf{u}_k) = -\begin{bmatrix} |\bar{\mathbf{b}}_{21}\mathbf{u}_k| & 0 \\ 0 & |\bar{\mathbf{b}}_{22}\mathbf{u}_k| \end{bmatrix} \mathbf{sign}(\mathbf{z}_1).$$

Hence, (4.17) is rewritten as

$$\dot{\mathbf{z}}_2 = \bar{\mathbf{f}}_2(\mathbf{x}_1, \boldsymbol{\chi}, \mathbf{w}, \mathbf{y}_r) - \mathbf{B}'(\boldsymbol{\chi}_1, \mathbf{w}, \mathbf{u})\mathbf{sign}(\mathbf{z}_2) \tag{4.20}$$

where

$$\mathbf{B}'(\boldsymbol{\chi}_1, \mathbf{w}, \mathbf{u}) = \begin{bmatrix} |\bar{\mathbf{b}}_{21}\mathbf{u}_k| & 0 \\ 0 & \bar{\mathbf{b}}_{22}\mathbf{u}_k| \end{bmatrix}$$

which guarantees a sliding mode on the surface $\mathbf{z}_2 = 0$ in a finite time. Under the condition

$$\left|\bar{f}_{21}(\mathbf{x}^1, \boldsymbol{\chi}, \mathbf{w}, \mathbf{y}_r, \dot{\mathbf{y}}_r)\right| < \left|\bar{\mathbf{b}}_{21}\mathbf{u}_k\right| \text{ and } \left|\bar{f}_{22}(\mathbf{x}^1, \boldsymbol{\chi}, \mathbf{w}, \mathbf{r}, \dot{\mathbf{r}})\right| < \left|\bar{\mathbf{b}}_{21}\mathbf{u}_k\right|, \quad \forall t. \tag{4.21}$$

The sliding dynamics, in the tracking errors variables z_1 and z_2 (4.8), is governed by the second order linear system

$$\begin{aligned} \dot{z}_1 &= -k_1 z_1 \\ \dot{z}_2 &= -k_2 z_2 \end{aligned}$$

with desired eigenvalues $-k_1$ and $-k_2$.

Note that some of the available input vectors, presented in Table 1 may satisfy the condition (4.18), at the same time; we select the input vector that maximizes $\left\|\bar{\mathbf{B}}_2(\boldsymbol{\chi}^1, \mathbf{w})\mathbf{u}_k\right\|$, in order to increase the sliding motion stability margin given by (4.21).

4.6.2 Stability Analysis

In order to analyze stability, let propose the following candidate Lyapunov function

$$V = \frac{1}{2}\mathbf{e}^\top\mathbf{e} + \frac{1}{2}\sum_{i=1}^{3}\mathbf{w}_i^\top\Gamma_i\mathbf{w}_i + \frac{1}{2}\mathbf{z}_1^\top\mathbf{R}\mathbf{z}_1 + \frac{1}{2}\mathbf{z}_2^\top\mathbf{z}_2 \tag{4.22}$$

where $\mathbf{R}$ is a symmetric positive definite matrix and $\mathbf{e} = [e_1, e_2, e_3]^\top$ Assuming that the modeling error terms are zeros and differentiating (4.22) along the trajectories of (2.13), (4.2), (4.6), (4.16) and (4.20), it yields

$$\dot{V} = -\mathbf{e}^\top\mathbf{Q}\mathbf{e} - \mathbf{z}_1\mathbf{S}\mathbf{z}_1 + \mathbf{z}_2^\top\left(2\mathbf{R}\mathbf{z}_1 + \bar{\mathbf{f}}_2 - \mathbf{B}'\mathbf{sign}(\mathbf{z}_2)\right)$$

with $\mathbf{Q} = -(\mathbf{A} + \mathbf{A}^\top)$ and $\mathbf{S} = -(\mathbf{R}\mathbf{K} + \mathbf{K}^\top\mathbf{R})$ as positive definite matrices. Additionally, assume that the following bound is satisfied

$$\left\|\bar{\mathbf{f}}_2 + 2\mathbf{R}\mathbf{z}_1\right\|_2 \leq q_0, \text{and } q_0 > 0$$

If the matrix $\mathbf{B}'$ satisfies

$$\left|\bar{\mathbf{b}}_{2,i}\mathbf{u}_k\right| \geq q_0 + d_0, \quad d_0 > 0, \ i = 1, 2$$

then

$$\mathbf{z}_2^\top\mathbf{B}'\mathbf{sign}(\mathbf{z}_2) + \mathbf{z}_2^\top\left[2\mathbf{R}\mathbf{z}^1 + \bar{\mathbf{f}}_2\right] \leq -\left(q_0 + d_0\right)\left\|\mathbf{z}_2\right\|_1 + q_0\left\|\mathbf{z}_2\right\|_2 \leq -d_0\left\|\mathbf{z}_2\right\|_1$$

which shows that the time derivative $\dot{V}$ satisfies the following inequality

$$\dot{V} \leq -\mathbf{e}^{\top}\mathbf{Q}\mathbf{e} - \mathbf{z}_1^{\top}\mathbf{S}\mathbf{z}_1 - d_0 \|\mathbf{z}_2\|_1$$

Using Lemma 3.1, we conclude that plant output $\mathbf{y}_P = [\chi_1, \Psi]$ tracks the reference signal $\mathbf{y}_r = [\omega_r, \varphi_r]^{\top}$

For the three control strategies developed in this chapter, it is possible to prove that vector $[\mathbf{e}^{\top}, \mathbf{z}^{\top}, \mathbf{w}_1^{\top}, \mathbf{w}_2^{\top}, \mathbf{w}_3^{\top}]^{\top}$ converges into a bounded ball, when the modeling errors are not zero but bounded. The proof is straight forward (see Lemma 3.2).

4.7 Simulations

In this section, we present the results obtained using the identification scheme and the control laws proposed above. The nominal values of the induction motor parameters are given in the next table:

Parameter	Value	Description
R_s	14Ω	Stator Resistance
L_s	$400mH$	Stator self Inductance
M	$377mH$	Mutual Inductance
R_r	10.1Ω	Rotor Resistance
L_r	$412.8mH$	Rotor self Inductance
n_p	2	Number of pairs of poles
J	$0.01Kgm$	Inertia Momentum

The design parameters for the fluxes observer are $l_1, l_2 = 3500$; for the neural network, we selected $a_1 = 100$, $a_2 = a_3 = 500$, $\beta = 0.1$, $\Gamma_1^{-1} = diag\{500, 500, 500\}$, $\Gamma_2 = \Gamma_3 = diag\{500, 500, 50\}$. In order to test the proposed scheme performance, a variation of 2 Ohm per second is added to the rotor resistance, in addition we include a square torque disturbance (see Figure 4.3 and Figure 4.4).

4.7.1 Neural Block Controller Simulation

The results for velocity and flux are presented in Figure 4.5 and Figure 4.6, respectively. As can be seen, the performance of the proposed scheme is very satisfactory. Figure 4.7 shows

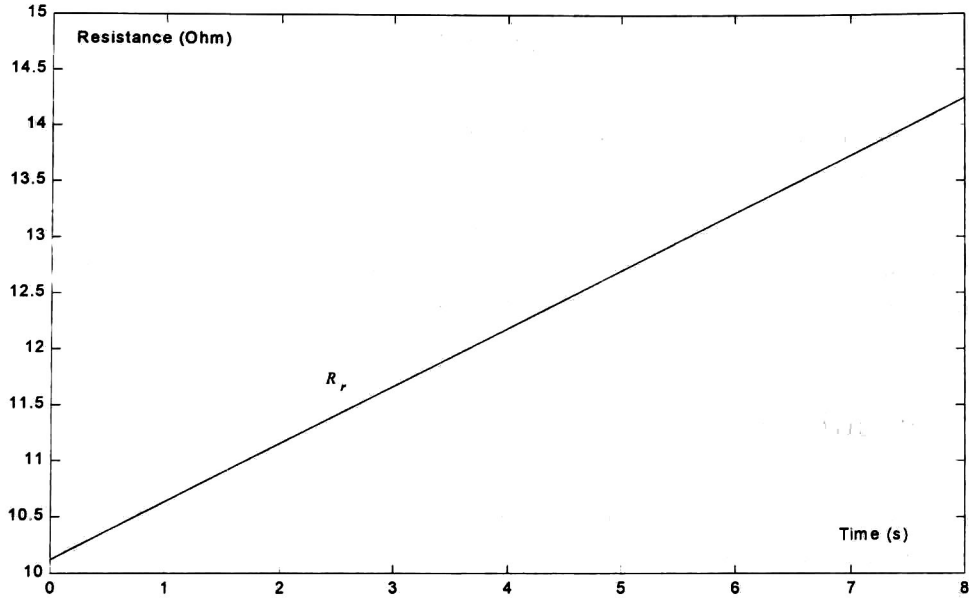


Figure 4.3: Rotor resistance R_r.

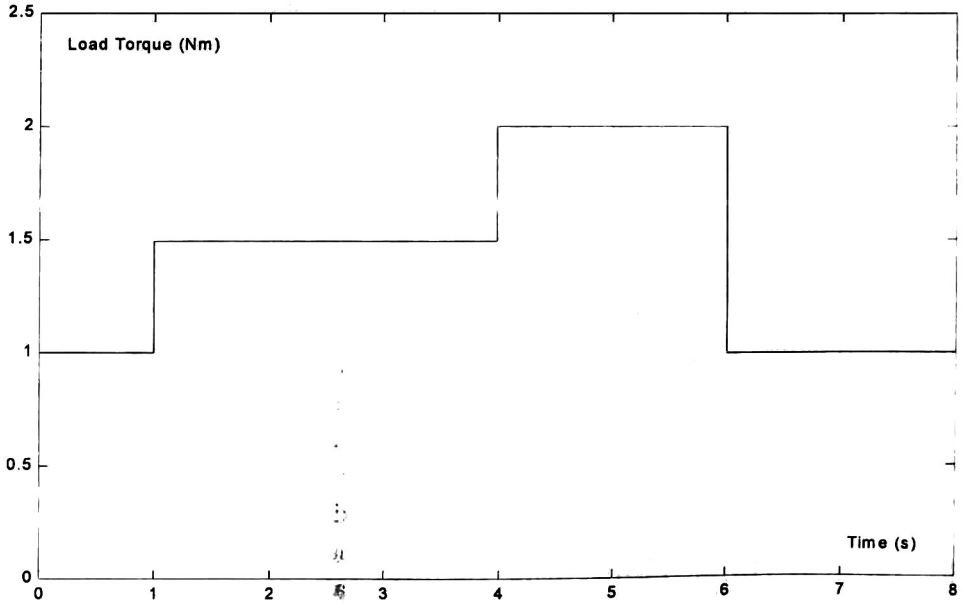


Figure 4.4: Load torque T_L

the flux observer and flux identifier response. The current in phase α an its estimated can be compared in Figure 4.8. Figure 4.9 is a phase portrait of the stator currents. Finally, the identifier parameters are plotted in Figure 4.10.

4.7.2 SSM Controller Simulation

Figure 4.11 shows the speed behavior, using the SSM controller. Although the robustness of this control law is very good, the currents harmonics may be unacceptable, as can be appreciated in Figure 4.12.

4.7.3 Simulation of Neural Block Controller with Discrete Input Constrains

Figures 4.13 and 4.14 show the performance of this control strategy. Figure 4.15 is a phase portrait of the stator currents, it can be verified that the harmonics are low.

4.7.4 Modification for robustness improving

The robustness is improved by substituting $\mathbf{z}^1 = \begin{bmatrix} \chi_1 - \omega_r \\ \Psi - \varphi_r \end{bmatrix}$ instead of $\mathbf{z}^1 = \begin{bmatrix} x_1 - \omega_r \\ \varphi - \varphi_r \end{bmatrix}$ in the control laws. Although, the control law is more robust than the explained in Sections 4.4, 4.5 and 4.6, it does not fit the stated stability analysis, so this work are left as a future research topic. Figures 4.16-4.20 show the improved robustness of this modification.

4.8 Conclusions

Based on recurrent neural networks and VSC methodology, three control strategies for induction motors are presented. The stability, for both the identifier and the controller, is analyzed, and it is proved that the proposed control laws forces the closed loop trajectory to converge and stay in a manifold, which guarantees that the tracking error tends to zero. The robustness of these control scheme is tested in presence of different kind of disturbances such as load torque variations and changes on the induction motor parameters. The simulation results are very encouraging, Additionally [36] and [13] are two publications, which resume the results of this chapter..

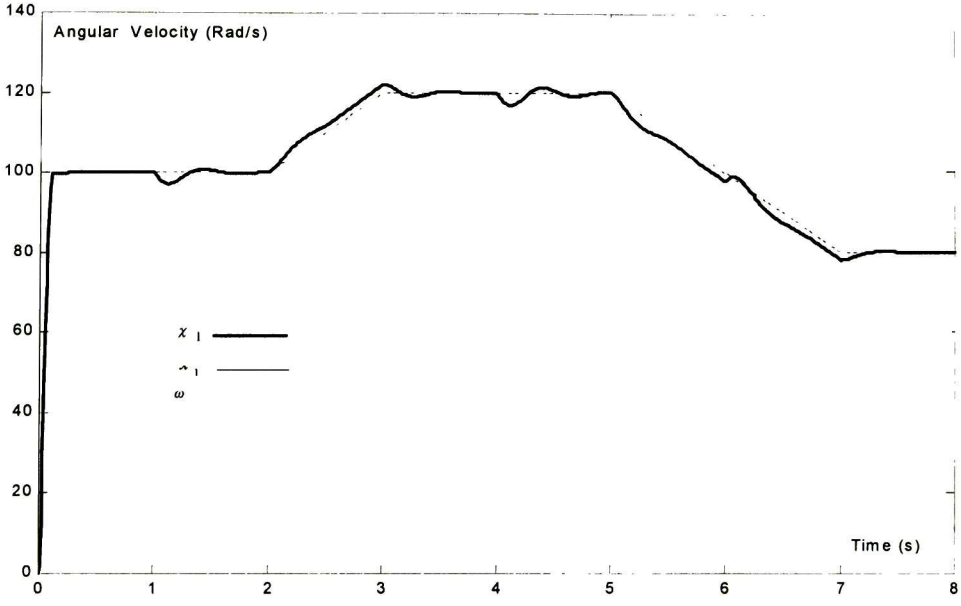


Figure 4.5: NBC; Real speed χ_1, reference speed ω_r and neural speed x_1.

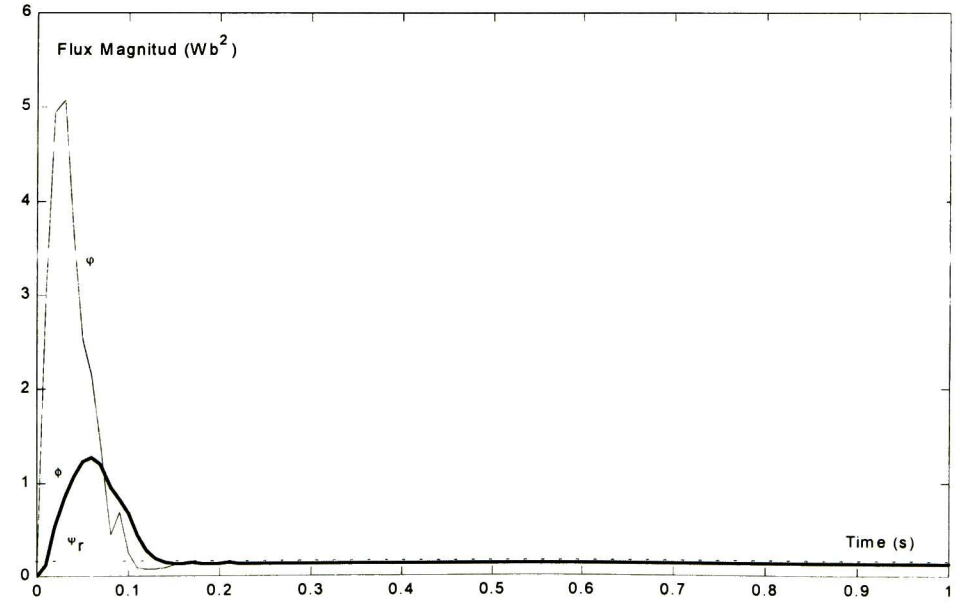


Figure 4.6: NBC; Flux magnitude ϕ neural flux magnitud φ and flux magnitud reference φ_r.

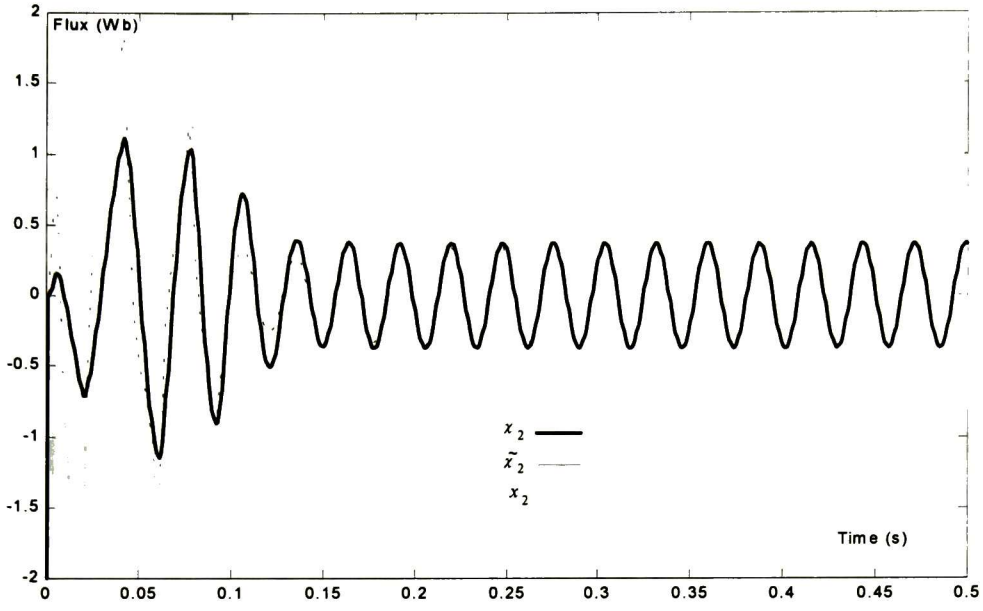


Figure 4.7: NBC; Flux χ_2, flux observer estimation $\hat{\chi}_2$ and flux identifier estimation x_2.

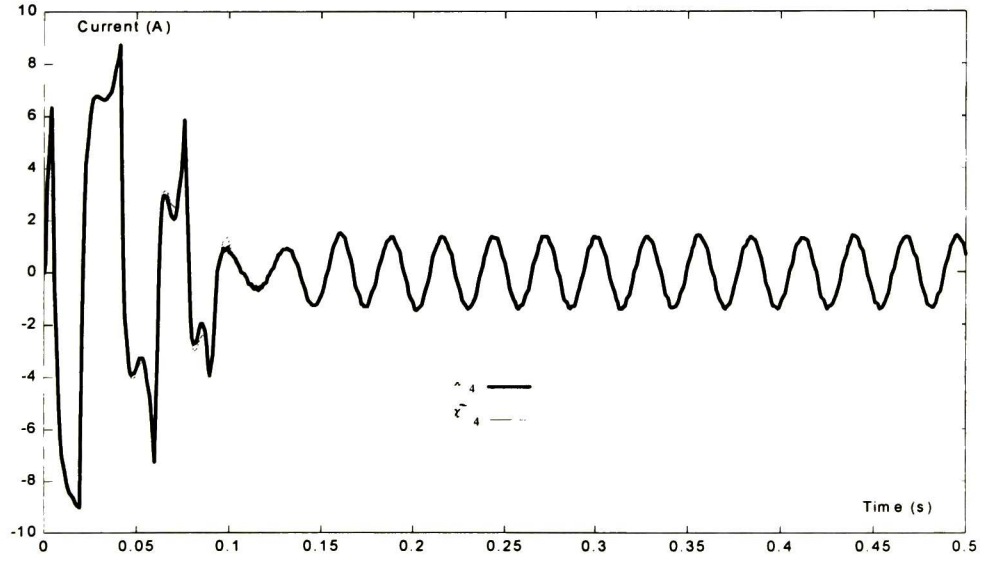


Figure 4.8: NBC; Stator current χ_4 and its estimated $\tilde{\chi}_4$.

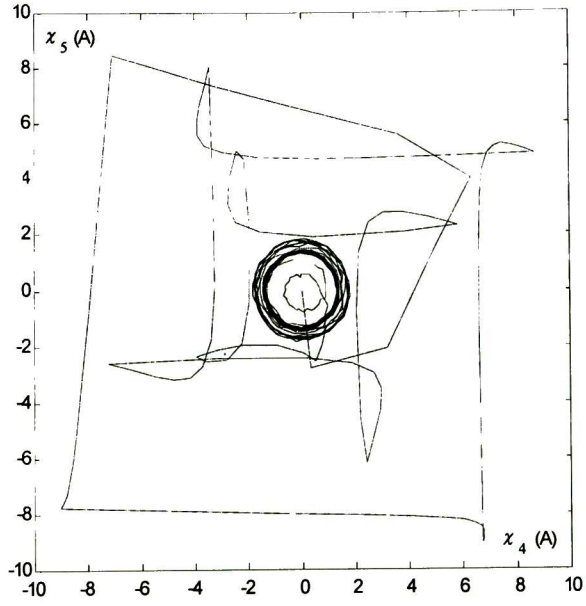


Figure 4.9: NBC; Stator currents phase portrait.

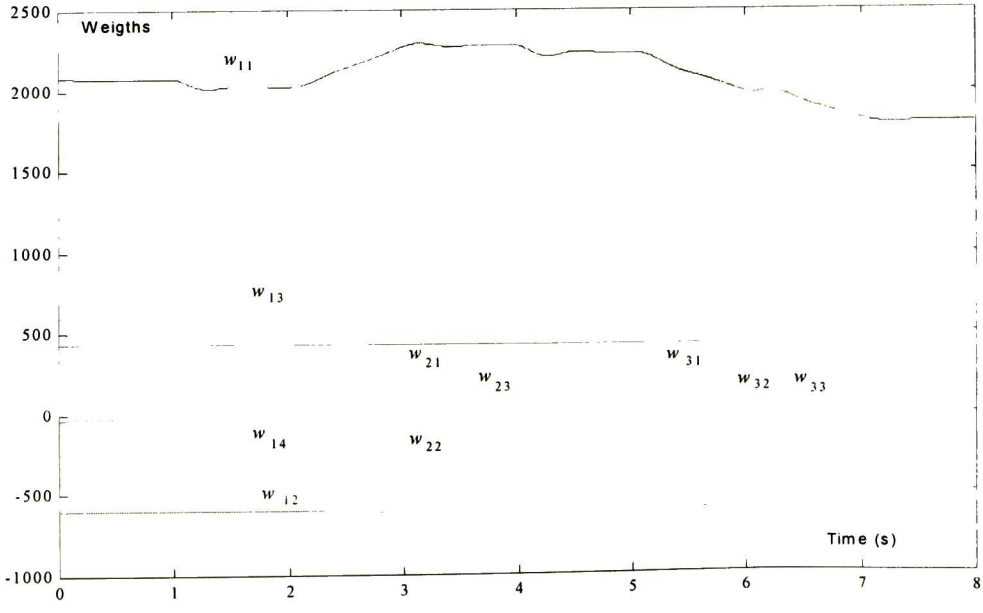


Figure 4.10: NBC; RHONN weights

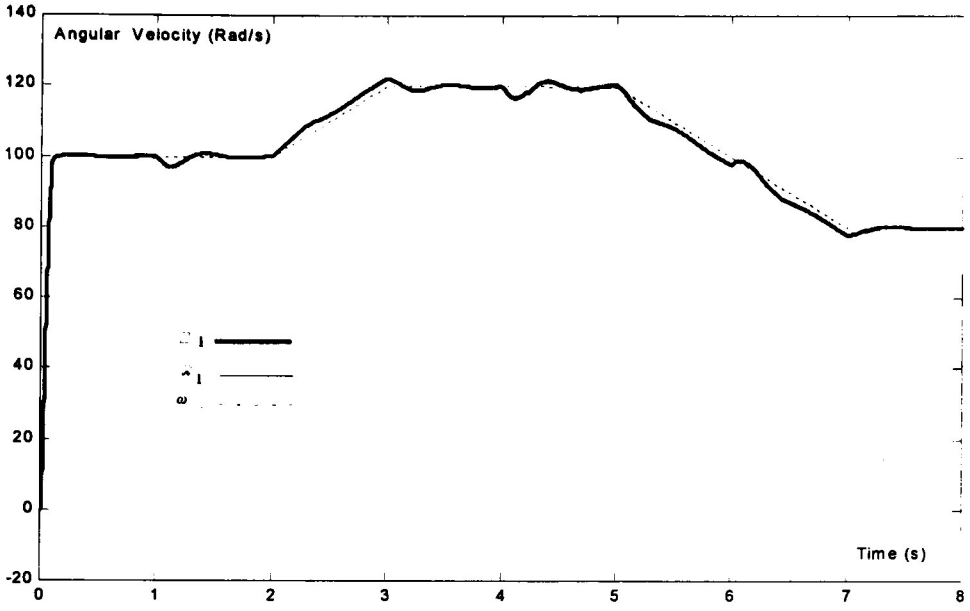


Figure 4.11: SSM control; Real speed χ_1, speed reference ω_r, neural speed x_1.

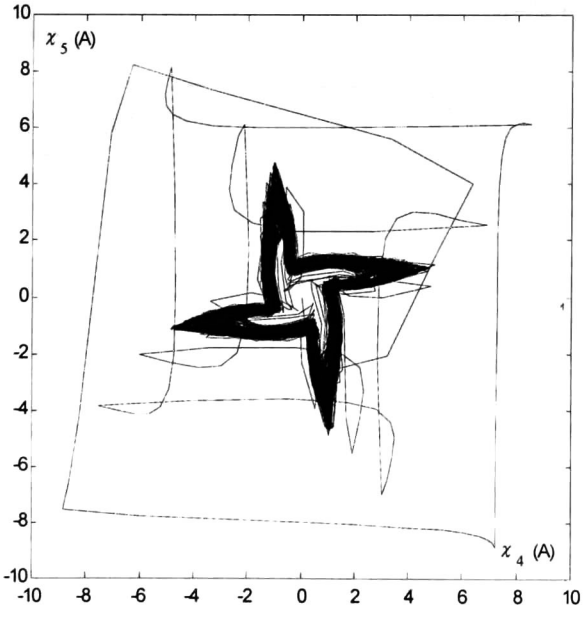


Figure 4.12: SSM control; Stator currents phase portrait.

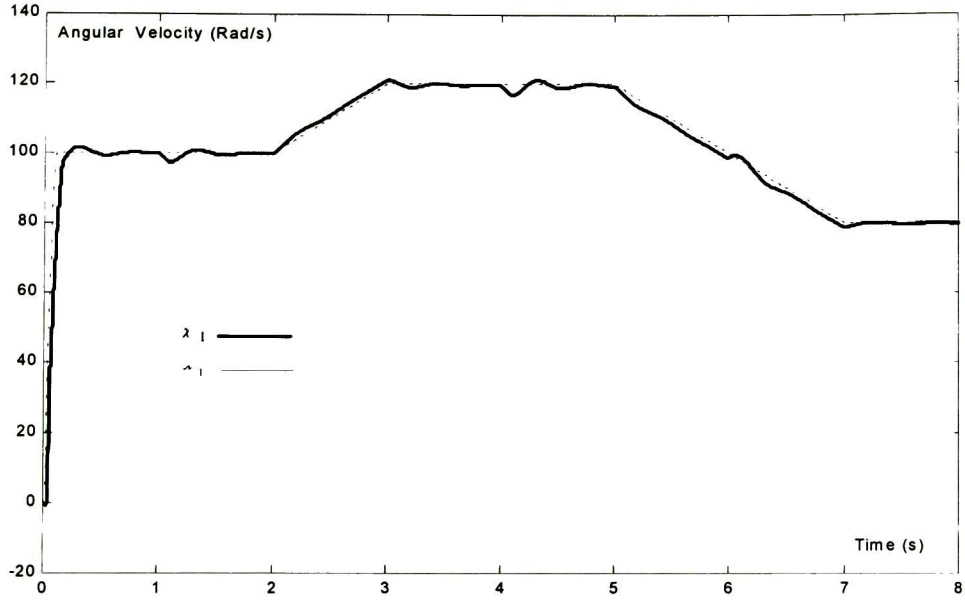


Figure 4.13: NBC with input constrains; Real speed χ_1, speed reference ω_r, neural speed x_1.

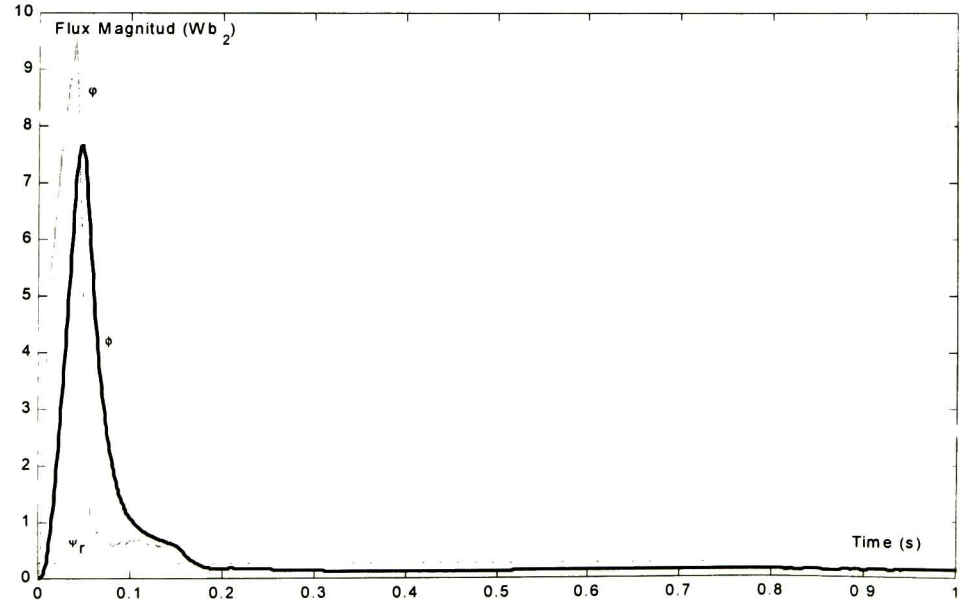


Figure 4.14: NBC with input constrains: Flux magnitude ϕ neural flux magnitud φ and flux reference φ_r.

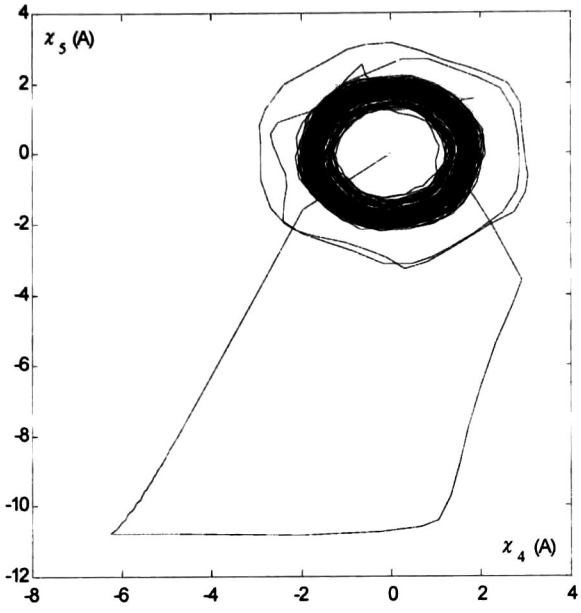


Figure 4.15: NBC with input constrains; Stator currents phase portrait.

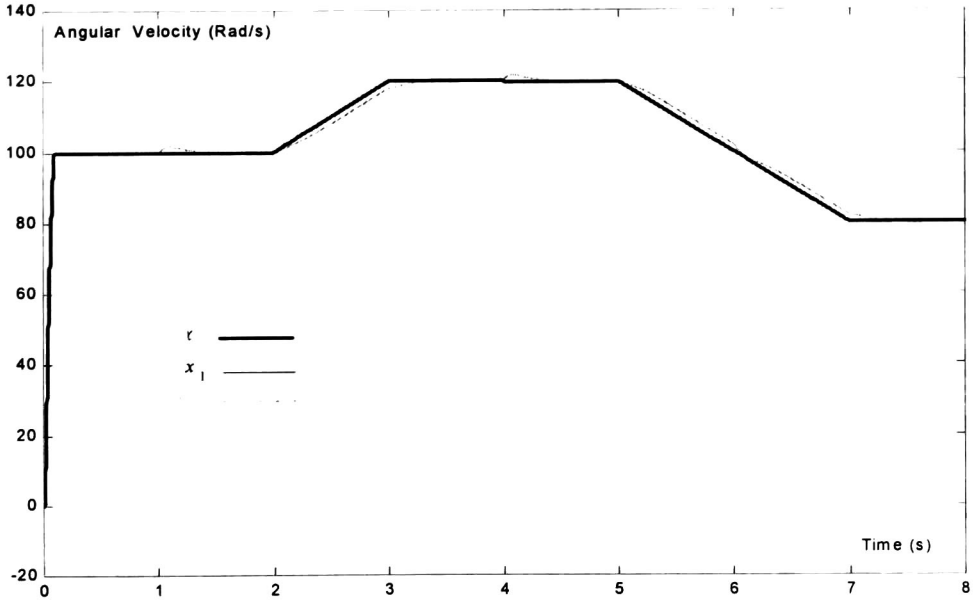


Figure 4.16: Robust NBC; Real speed χ_1, speed reference ω_r, speed identifier estimation x_1.

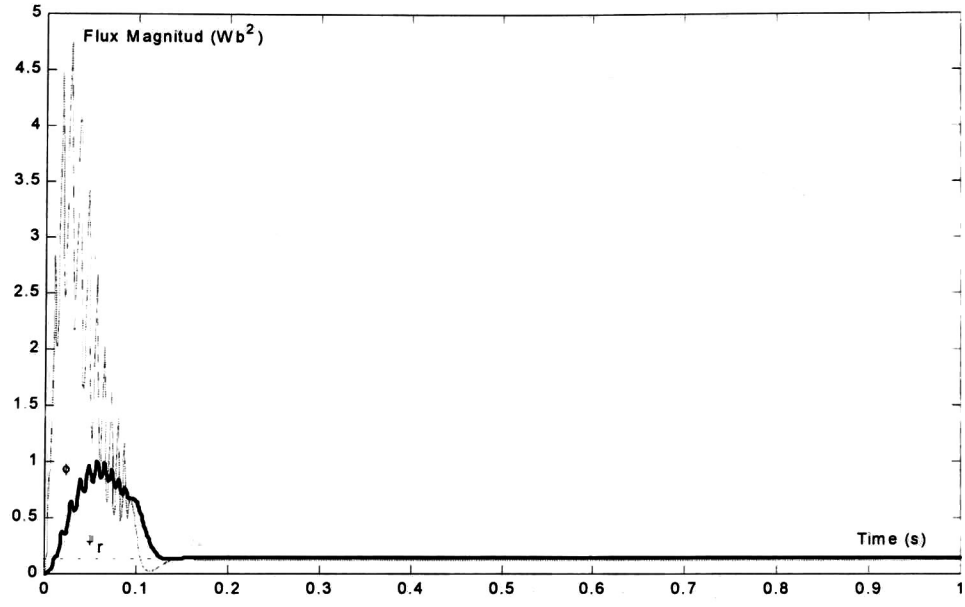


Figure 4.17: Robust NBC: Flux magnitude ϕ neural flux magnitud φ and flux reference φ_r.

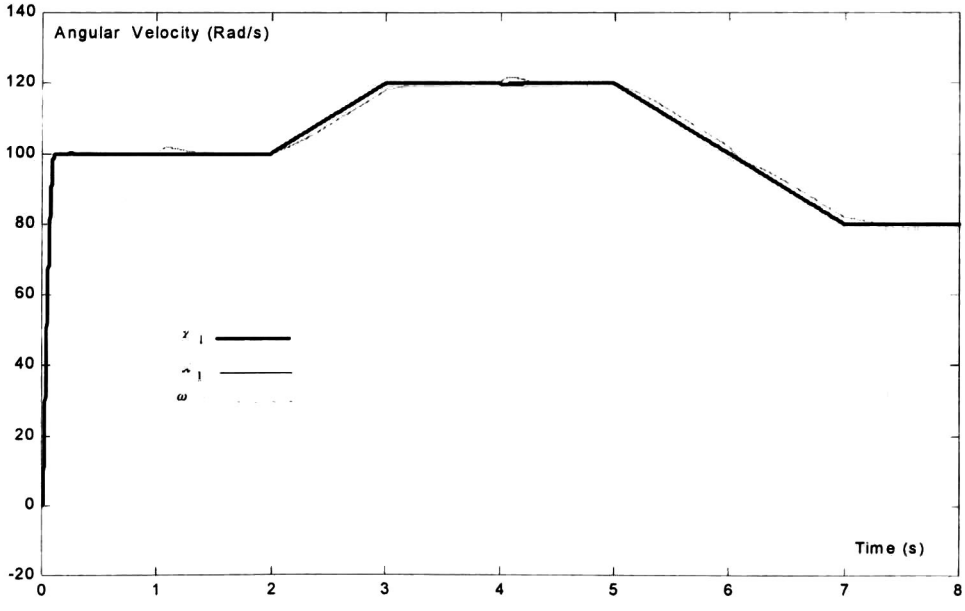


Figure 4.18: Robust SSM control; Real speed χ_1, speed reference ω_r, neural speed x_1.

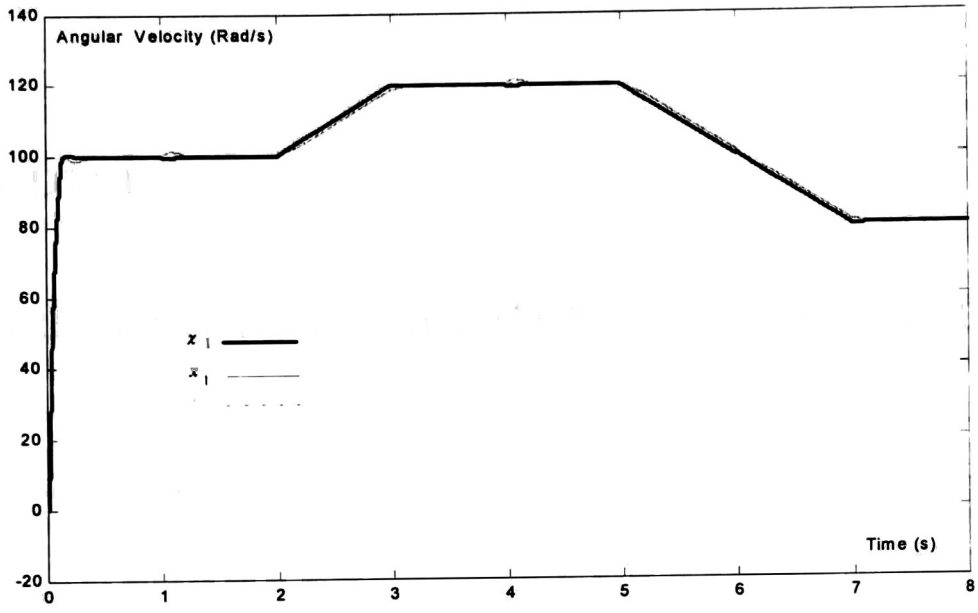


Figure 4.19: Robust NBC with input constrains; χ_1, ω_r and x_1.

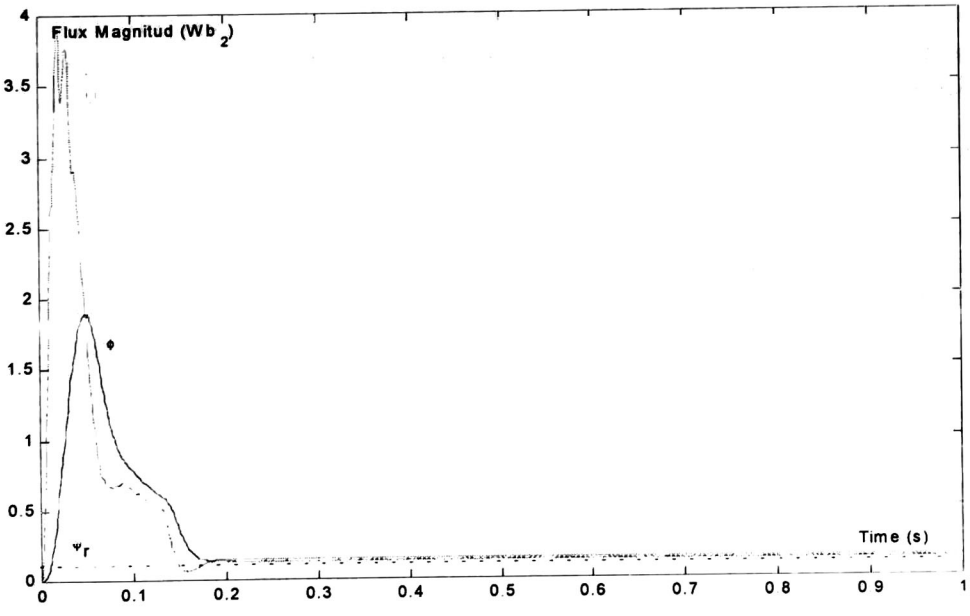


Figure 4.20: Robust NBC with input constrains: ϕ, φ and φ_r.

Chapter 5

Synchronous Generator Neural Control

Modern human society needs an ever-increasing supply of electrical power. Very complex power systems have been built to satisfy this increasing demand. The trend in electric power production is toward an interconnected network of transmission lines linking generators and loads into large integrated systems. Successful operation of a power system depends largely on the engineer's ability to provide reliable and interrupted service to the loads; ideally the loads must be fed at constant frequency at all times.

The first requirement of a reliable service is to keep the synchronous generators running in parallel and with adequate capacity to meet the load demand. If at any time a generator loses synchronism with the rest of the system, significant voltage and current fluctuations may occur and transmission lines may be automatically tripped by their relays at undesired locations.

Feedback linearization was one of the early strategies to be explored, with applications proposed to both single and multimachine systems, even with output feedback and state observers [39], [27], [24], [61] and [41]. Robustness issues, both against parameter uncertainties and not modelled dynamics are still open, as well as stability analysis for the output feedback case. Others recent works on energy related design techniques, have been developed for the well-known single machine third-order flux decaying [45]; most of these results are based on the application of damping injection controller [2], [14]. As done in Chapter 4, Block Control and Neural Networks are combined to obtain a robust control law, capable to reject disturbances which affect the system.

5.1 Mathematical Model

Commonly an electric power network consists of N generators connected by lossless transmission lines. It is assumed that the mechanical input power of each generator is constant. The machine model considered here is the classical flux-decay model (the one axis model) given in [1] and [15]; exciters and governors are not included in this model. Thus, we have for $k = 1, ..., N$:

$$\begin{aligned}
\dot{\delta}_k &= \Delta\omega_k \\
M_k\dot{\omega}_k &= P_{mk} - P_{ek} - D_k\omega_k - \frac{M_k}{M_T}P_{COI} \\
T'_{d0k}\dot{E}_{qk} &= \frac{X_{dk} - X'_{dk}}{X'_{dk}}V_{k+n}\cos(\delta_k - \theta_{k+n}) + E_{fdk} - \frac{X_{dk}}{X'_{dk}}E'_{qk}
\end{aligned}$$

where subscript k relates to the generator number, δ_k is the load angle, $\Delta\omega_k = \omega_k - \omega_s$ is the speed deviation, and E_{qk} is the quadrature axis internal voltage, P_{mk} is the mechanical power, E_{fdk} is the excitation field voltage, $P_{COI} = \sum_{k=1}^{N}(P_{mk} - P_{ek})$ and P_{ek} is the generated electrical power. Full details of the model and definitions of parameters can be found in [15] and [1].

For the single machine infinite bus (SMIB) case, the above model simplifies to the well-known third-order model [45], which is used in most transient stability studies in power systems. The dynamics of a single synchronous generator is described by the following equations [2]

$$\dot{\chi}_1 = \chi_2 \tag{5.1a}$$

$$\dot{\chi}_2 = -b_1\chi_3\sin(\chi_1) - b_2\chi_2 + P \tag{5.1b}$$

$$\dot{\chi}_3 = b_3\cos(\chi_1) - b_4\chi_3 + E + u \tag{5.1c}$$

The state variables of this system are the load angle χ_1, the shaft speed deviation from the synchronous speed χ_2, and the quadrature axis internal voltage χ_3. The inputs $P = \frac{\omega_s P_m}{2H}$ and $E = \frac{E_{fd}}{T'_{d0}}$ are held constant, and a supplementary signal u is added to the field voltage which represents the control input. The coefficients b_i, $i = 1, ..., 4$ are positive.

5.2 Neural Model for Synchronous Generators

Notice that the model (5.1) has relative degree 3, so we cannot apply the control scheme used for the induction motor in Chapter 4. On other hand, the general neural block control of Section 3.2 may be inadequate due to the uncertainties involved in such controller. Nevertheless, we can take advantage from the model (5.1a), it is has not parameters which may vary with time, so it is valid to consider no adaptive parameter for its neural identifier state. Then, some slight changes are made to the identification scheme of Chapter and to the control strategy of Chapter 3, in order to fit them to this generator model.

So, let assume that the model (5.1a)-(5.1b) has the RHONN representation given by

$$\dot{\chi}_1 = \chi_2 \tag{5.2a}$$
$$\dot{\chi}_2 = -\chi_2 + w_{21}^* S(\chi_2) + w_{22}^* S(\chi_1)S(\chi_3) + P + \nu_2(t) \tag{5.2b}$$

where $\mathbf{w}_2^* = [w_{21}^*, w_{22}^*]^\top$ is the optimal weight vector, which is constant and unknown, $\boldsymbol{\rho}_2 = [S(\chi_2), S(\chi_1)S(\chi_3)]^\top$ is the high order term vector and $\nu_2(t)$ is the modelling error term.

For this application some modifications are done with respect to the identification scheme used before. Based on the neural model (5.2a)-(5.2b), we propose the next neural model to identify the synchronous generator (5.1)

$$\dot{x}_1 = -a_1(x_1 - \chi_1) + x_2 \tag{5.3}$$
$$\dot{x}_2 = -a_2 x_2 + w_{21} S(\chi_2) + w_{22} S(\chi_1)S(\chi_3) + P$$

According to the neural structure explained in Chapter 2; $\boldsymbol{\chi} = [\chi_1,\ \chi_2,\ \chi_3]^\top$ $\mathbf{x} = [x_1,\ x_2]^\top$ $\mathbf{w}_2 = [w_{21},\ w_{22}]^\top$ $\boldsymbol{\rho}_2 = [S(\chi_2),\ S(\chi_1)S(\chi_3)]^\top$

Now, assuming a zero modeling error term, the identification error dynamics is

$$\dot{\mathbf{e}} = \mathbf{A}\mathbf{e} + \mathbf{b}(\tilde{\mathbf{w}}_2^\top \boldsymbol{\rho}_2)$$

where $\mathbf{e} = [e_1\ \ e_2]^\top$ $\mathbf{A} = \begin{bmatrix} -a_1 & 1 \\ 0 & -a_2 \end{bmatrix}$.and $\mathbf{b} = [0\ 1]^\top$ Let define the following candidate Lyapunov function

$$V = \frac{1}{2}\mathbf{e}^\top\mathbf{e} + \frac{1}{2}\tilde{\mathbf{w}}_2^\top \Gamma_2 \tilde{\mathbf{w}}_2^\top$$

by selecting $a_1 a_2 > \frac{1}{4}$, $\mathbf{Q} = -(\mathbf{A}^\top + \mathbf{A})$ is a positive definite matrix, then

$$\dot{V} = -\mathbf{e}^\top\mathbf{Q}\mathbf{e} + e_2\tilde{\mathbf{w}}_2^\top\boldsymbol{\rho}_2 + \tilde{\mathbf{w}}_2^\top\Gamma_2\dot{\tilde{\mathbf{w}}}_2^\top$$

Recalling Chapter 2 and using the adaptive law

$$\dot{\mathbf{w}}_2 = e_2\Gamma_2^{-1}\boldsymbol{\rho}_2 \tag{5.4}$$

the convergence to zero of $\mathbf{e}$ and the boundness of $\mathbf{w}_2$, are assured.

If the modeling error term is no zero, the σ-modification (2.14) is applied to guarantee the identification and parametric errors boundness; see Lemma 2.2.

5.3 Neural Block Control

The goal of the proposed scheme is to force the synchronous generator to track a specified reference. In order to achieve this tracking, we develop a control law based on the sliding modes technique. Let define δ_r as the desired constant value for the load angle. Then according to Chapter 3 the plant output is $\mathbf{y}_P = \chi_1$, the neural output is $\mathbf{y}_N = x_1$ and the reference signal is $\mathbf{y}_r = \delta_r$.

The tracking error is given by

$$z_1 = x_1 - \delta_r \tag{5.5}$$

and the dynamics for the new variable z_1 can be obtained from (5.3) and (5.5) as

$$\dot{z}_1 = -a_1 e_1 + x_2 = -k_1 z_1 + z_2 \tag{5.6}$$

with the new variable

$$z_2 = k_1 z_1 - a_1 e_1 + x_2$$

whose dynamics is given by

$$\dot{z}_2 = k_1(-a_1 e_1 + x_2) - a_1(a_1 e_1 + x_2) - a_2 x_2 + w_{21} S(\chi_2) + w_{22} S(\chi_1) S(\chi_3) + P = -k_2 z_2 + z_3$$

with

$$z_3 = f_2(\boldsymbol{\chi}, \mathbf{x}) + w_{22} S(\chi_1) S(\chi_3)$$

where $f_2(\boldsymbol{\chi}, \mathbf{x}) = k_2 z_2 + k_1(-a_1 e_1 + x_2) - a_1(a_1 e_1 + x_2) - a_2 x_2 + w_{21} S(\chi_2) + P$

Then the time derivative of z_3 is expressed by

$$\begin{aligned} \dot{z}_3 &= \frac{\partial f_2(\boldsymbol{\chi})}{\partial \boldsymbol{\chi}} \dot{\boldsymbol{\chi}} + \frac{\partial f_2(\mathbf{x})}{\partial \mathbf{x}} \dot{\mathbf{x}} + \dot{w}_{22} S(\chi_1) S(\chi_3) + w_{22} S(\chi_3) \frac{\partial S(\chi_1)}{\partial \chi_1} \dot{\chi}_1 \\ &\quad + w_{22} S(\chi_1) \frac{\partial S(\chi_3)}{\partial \chi_3} (b_3 \cos(\chi_1) - b_4 \chi_3 + E + u) \end{aligned} \tag{5.7}$$

Taking into account that the control should be bounded, $|u| \leq u_0$, $u_0 > 0$, we propose the following discontinuous control law:

$$u = -u_0 sign(w_{22} S(x_1) \frac{\partial S(\chi_3)}{\partial \chi_3}) sign(z_3) \tag{5.8}$$

Then the closed system (5.7), (5.8) becomes

$$\dot{z}_3 = f_3 - u_0 |w_{22} S(x_1) \frac{\partial S(\chi_3)}{\partial \chi_3}| sign(z_3) \tag{5.9}$$

where $f_3 = \frac{\partial f_3(\chi)}{\partial \chi}\dot{\chi} + \frac{\partial f_3(\mathbf{x})}{\partial \mathbf{x}}\dot{\mathbf{x}} + \dot{w}_{22}S(\chi_1)S(\chi_3) + w_{22}S(\chi_3)\frac{\partial S(\chi_1)}{\partial \chi_1}\dot{\chi}_1 + w_{22}S(\chi_1)\frac{\partial S(\chi_3)}{\partial \chi_3}(b_3\cos(\chi_1) - b_4\chi_3 + E)$.

If the following condition is satisfied

$$|f_3| > u_0|w_{22}S(x_1)\frac{\partial S(\chi_3)}{\partial \chi_3}|, \quad \forall t$$

a sliding motion on the nonlinear surface $z_3 = 0$ is guaranteed [58]. Then the control error sliding mode dynamics on this surface is described by the second order linear system

$$\begin{aligned} \dot{z}_1 &= -k_1 z_1 + z_2 \\ \dot{z}_2 &= -k_2 z_2 \end{aligned} \tag{5.10}$$

which is stable for $k_1 > 0$ and $k_2 > 0$. Hence, we can assure asymptotic convergence for the control errors. Notice that (5.10) can be rewritten as

$$\dot{\mathbf{z}}_1 = \mathbf{K}\mathbf{z}_1$$

where $\mathbf{z}_1 = \begin{bmatrix} z_1 \\ z_2 \end{bmatrix}$ $\mathbf{K} = \begin{bmatrix} -k_1 & 1 \\ 0 & -k_2 \end{bmatrix}$.

5.3.1 Stability Analysis

In order to analyze the stability, let propose the following candidate Lyapunov function

$$V = \frac{1}{2}\mathbf{e}^\top\mathbf{e} + \frac{1}{2}\mathbf{w}_2^\top\Gamma_2\mathbf{w}_2 + \frac{1}{2}\mathbf{z}_1^\top\mathbf{R}\mathbf{z}_1 + \frac{1}{2}z_3^\top z_3 \tag{5.11}$$

where $\mathbf{R}$ is a symmetric positive definite matrix and $\mathbf{e} = [e_1, e_2]^\top$ Assuming that the modeling error term is zero and differentiating (5.11) along the trajectories of (5.3), (5.4), (5.2) and (5.9), it yields

$$\dot{V} = -\mathbf{e}^\top\mathbf{Q}\mathbf{e} - \mathbf{z}_1\mathbf{S}\mathbf{z}_1 + z_2^\top\left(2\mathbf{b}\mathbf{R}\mathbf{z}_1 + \bar{f}_3 - u_0\left|w_{22}S(x_1)\frac{\partial S(\chi_3)}{\partial \chi_3}\right| sign(z_3)\right)$$

with $\mathbf{S} = -(\mathbf{R}\mathbf{K} + \mathbf{K}^\top\mathbf{R})$ are a positive definite matrices. Additionally, assume that the following bound is satisfied

$$\left|\bar{f}_3 + 2\mathbf{b}\mathbf{R}\mathbf{z}_{12}\right| \leq q_0, \text{and } q_0 > 0$$

If $u_0 \left| w_{22} S(x_1) \frac{\partial S(\chi_3)}{\partial \chi_3} \right|$ satisfies

$$u_0 \left| w_{22} S(x_1) \frac{\partial S(\chi_3)}{\partial \chi_3} \right| \geq q_0 + d_0, \quad d_0 > 0, \; i = 1, 2$$

then

$$z_3 u_0 \left| w_{22} S(x_1) \frac{\partial S(\chi_3)}{\partial \chi_3} \right| sign(z_3) z_3 + z_3 \left[\bar{f}_3 + \mathbf{2bRz}_1 \right] \leq -(q_0 + d_0) |z_3| + q_0 |z_3| \leq -d_0 |z_3|$$

which shows that the time derivative $\dot{V}$ satisfies the following inequality

$$\dot{V} \leq -\mathbf{e}^\top \mathbf{Q} \mathbf{e} - \mathbf{z}_1^\top \mathbf{S} \mathbf{z}_1 - d_0 |z_3| \tag{5.12}$$

Using Lemma 3.1, we conclude that plant output $\mathbf{y}_P = \chi_1$ tracks the reference signal $\mathbf{y}_r = \delta_r$.

It is possible to prove that vector $[\mathbf{e}^\top, \mathbf{w}_2^\top, z_1, z_2, z_3]^\top$ converges into a bounded ball, when the modeling errors are not zero but bounded. The proof is straight forward (see Lemma 3.2).

5.4 Simulations

The parameters of the model (5.1), given on the next table, are taken from [1], and the operating conditions are as in [14].

Parameter	*Value (p.u)*
b_1	34.29
b_2	0.0
b_3	0.149
b_2	0.3341
P	28.22
E	0.2405

We analyze the response of the system to a short circuit generated by the connection of a small impedance between the machine's terminal and the ground; this impedance is disconnected after a certain time, called the clearing time. Then, the system goes back to its pre-disturbance state; during the fault occurrence, trajectories could diverge, if no control action is introduced. The largest time interval, before instability, is named the critical clearing time.

The design parameters for the neural network were selected as $a_2 = 10$, $\beta = 1$, $\Gamma_2^{-1} = diag\{200, 200\}$, and $k_1 = 5$ and $k_2 = 5$ were used for the control law. In order to test the

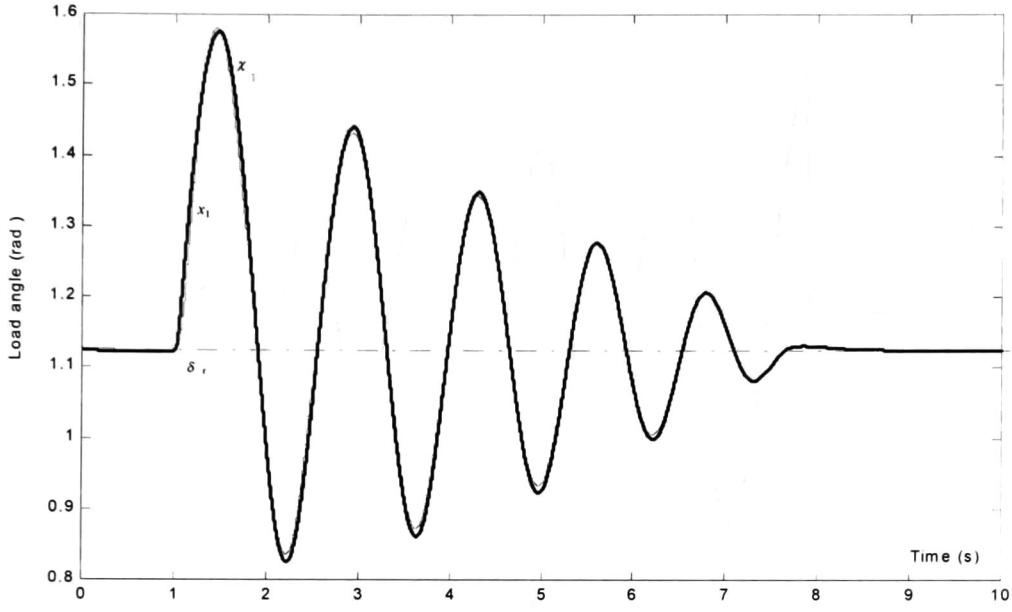


Figure 5.1: Plant angle (χ_1), identified angle (x_1) and reference angle (δ_r).

robustness of the proposed scheme, the described short circuit is added at $t = 1$, with an clearing time, $t_{cl} = 40ms$. The load angle is shown in Figure 5.1, In Figure 5.2 the angular velocity for the plant and for the neural identifier can be seen; the weights $\mathbf{w}_2$ are plotted in Figure 5.4, and the input signal is shown in Figure 5.5.

5.5 Conclusions

Similarly as done for induction motors, a neural identifier was proposed for the generator, but with some slight changes due to the relative degree of the plant. Nevertheless, the proposed control scheme shows good robustness under short circuit disturbance. The results of this chapter were presented in [12].

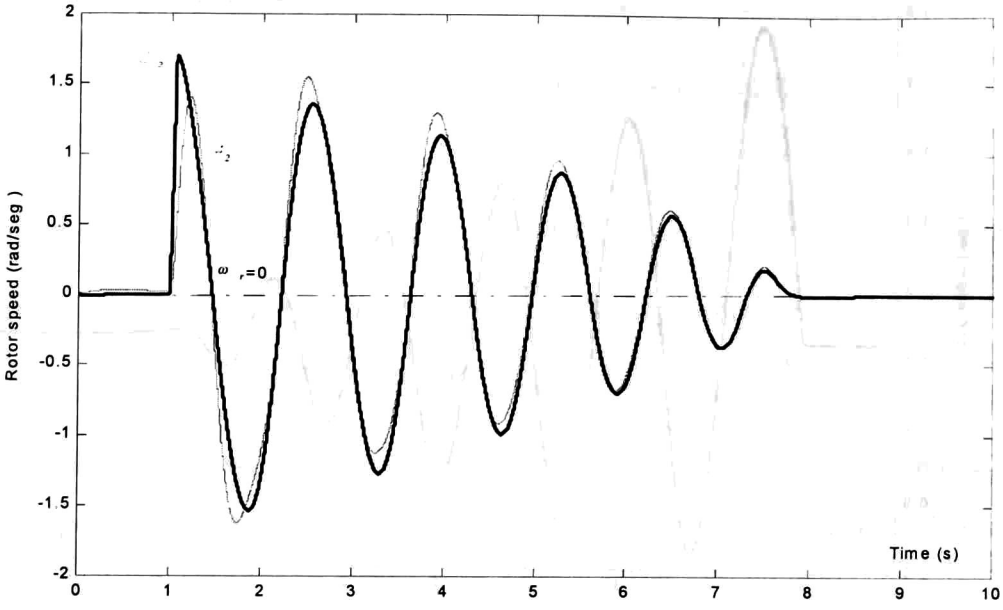


Figure 5.2: Plant speed (χ_2), identified speed (x_2) and reference speed (ω_r).

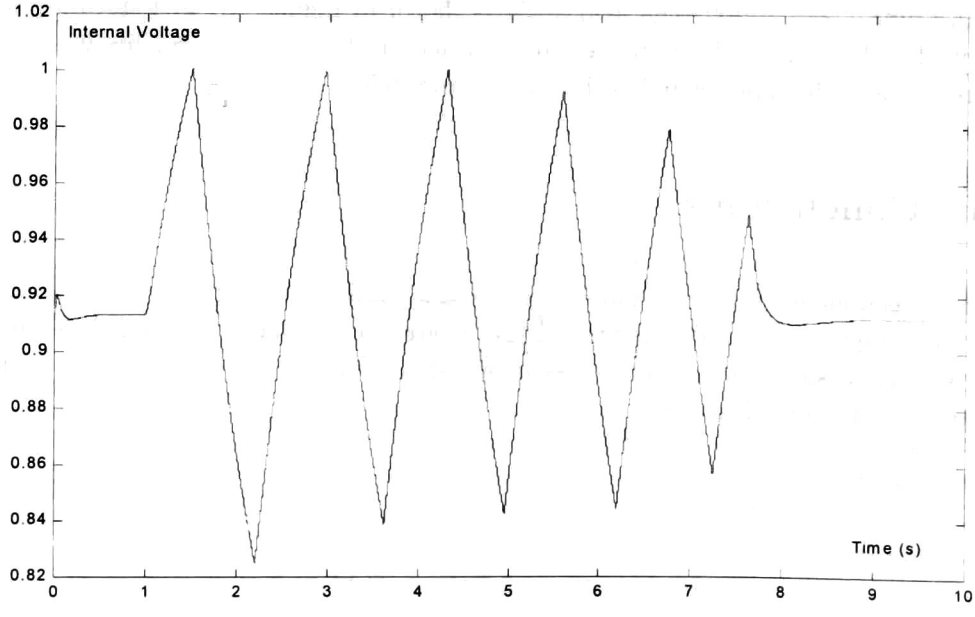


Figure 5.3: Internal Voltage χ_3

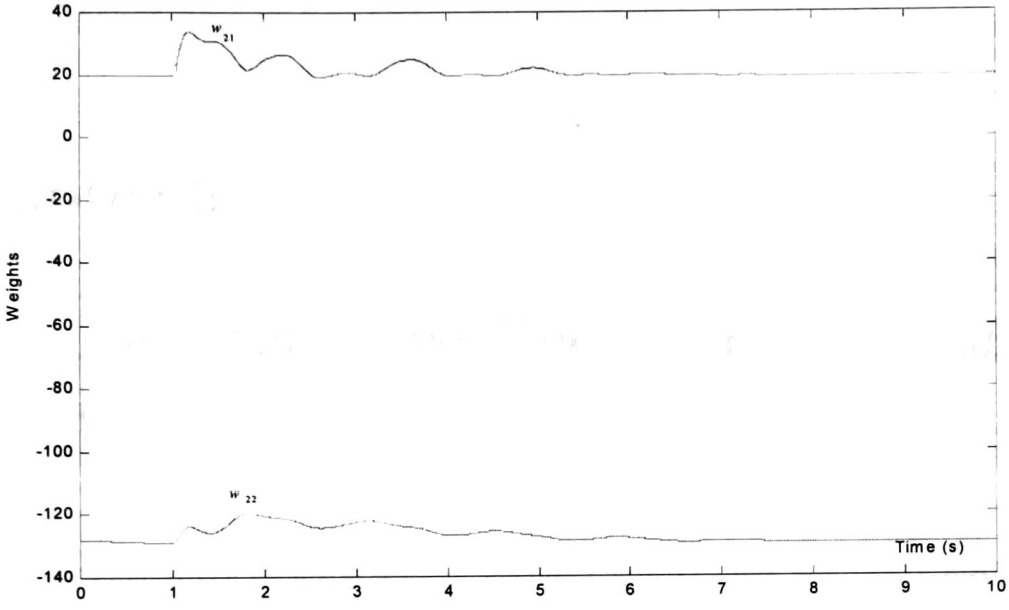


Figure 5.4: Weights for the angular speed x_2.

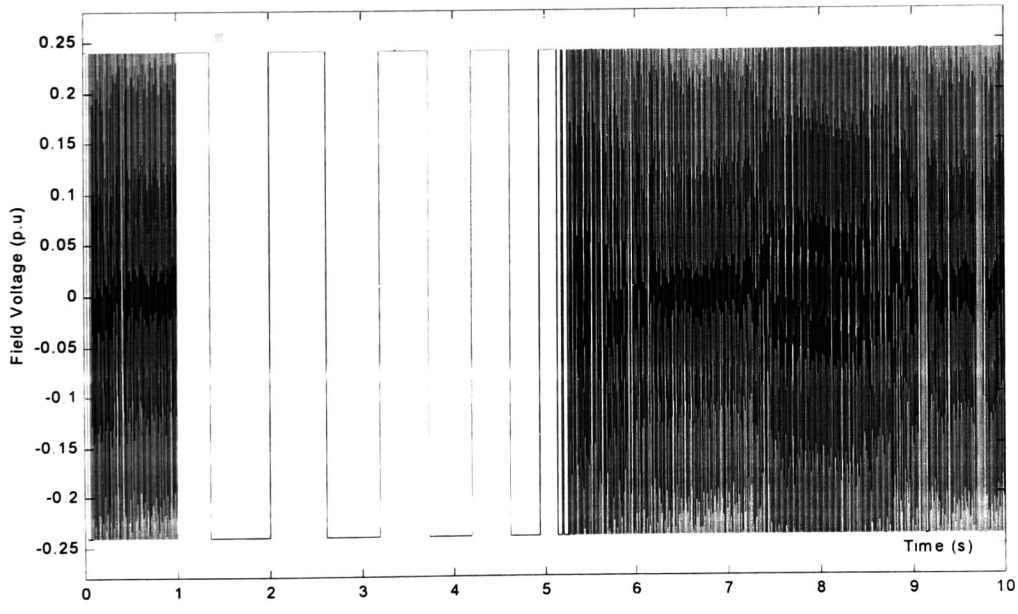


Figure 5.5: Control signal u.

Chapter 6

Variable Structure Recurrent Neural Networks

A challenger problem for nonlinear systems identification is to select a suitable structure for the identifier; recurrent neural networks offer the advantage of well approximating a nonlinear system to an arbitrarily accurate level [8], provided that the neural identifier has sufficiently large number of synaptic connections [29]. However, it is quite difficult to determine the number of sufficient synaptic connections to approximate a given dynamical system in general. If the neural identifier does not have enough synaptic connections, it is not possible to assure that the parameters converge to their optimal values, even using persistently excited inputs, and in many cases the identification error does not converge to zero. On the other hand, if there are too many synaptic connections, computational burden will be huge and the suggested solution becomes impractical.

In this chapter, to alleviate the aforementioned troublesome situation, the Variable Structure Neural Network (VSRNN) [49] are proposed, for continuous-time nonlinear dynamical systems identification. In this approach, an initial configuration for the neural identifier is first assumed. If a pre-specified error bound is not reached, more synaptic connections will be added, and another cycle of experimentation will begin, until the output performance of the network satisfies the pre-desired criterion. Complex chaotic systems are used to illustrate the capability and applicability of the proposed identification scheme.

6.1 Switching Systems

First it is convenient to define a switching system. Let the family of vector fields be $P = \{\mathbf{f}^\sigma(\mathbf{x}, \mathbf{u}, t), \mathbf{f}^\sigma : \Re^n \times \Re^m \rightarrow \Re^n, \sigma \in \Gamma\}$, where Γ is an index set, $\mathbf{f}^\sigma(0, 0, \cdot) = 0 \ \forall \sigma \in \Gamma$, and for

each $\sigma \in \Gamma$, $\mathbf{f}^\sigma$ is locally Lispchitz on $\mathbf{x}$ and uniformly on t [38]. Such vector fields are usually referred as subsystems.

Given the family P, consider a switched system,

$$\dot{\mathbf{x}} = \mathbf{f}^{s(t)}(\mathbf{x}, \mathbf{u}, t) \tag{6.1}$$

where $\mathbf{x} \in \Re^n$, $\mathbf{u} \in \Re^n$ $t \leq 0$, and s is the *switching signal*, defined as a time discontinuous function $[0, \infty) \rightarrow \Gamma$, and associated with the signal $s(t)$ there is a sequence of real numbers $\Upsilon = t_0 < t_1 < ... < t_k < ...$, called switching time sequence, and a index sequence $\Sigma = \sigma_0, \sigma_1, ..., \sigma_k, ...$, such that $s(t) = \sigma_k$ for all $t_k \leq t < t_{k+1}$.

Switched systems are of variable structure; they are a simple model of (the continuous portion) of hybrid systems [63]. So, $s(t)$ can be changed at any time by some higher process, such as a controller, computer, or human operators [4]. We focus specially on the *continuous* switched systems, which have the additional constraint that the switched subsystems agree at the switching time. More specifically, consider (6.1) with switching time sequence Υ and index sequence Σ, such that $\mathbf{f}^{s(t_{k-1})}(\mathbf{x}(t_k), \mathbf{u}(t_k), t_k) = \mathbf{f}^{s(t_k)}(\mathbf{x}(t_k), \mathbf{u}(t_k), t_k)$. So continuous switched systems require that the vector field is continuous over time.

6.2 VSRNN model

Define the dynamics of each neuron of a Variable Structure Recurrent Neural Networks (VSRNN) [49] by

$$\dot{x}_i = -a_i x_i + f_i^{s_i}(\mathbf{x}, \mathbf{u}, \mathbf{w}_i), \quad i = 1, ..., n. \tag{6.2}$$

For the $i-$th neuron, consider a family of functions: $P_i = \{f_i^\sigma(\mathbf{x}, \mathbf{u}, \mathbf{w}_i), f_i^\sigma : \Re^n \times \Re^m \times \Re^{L_i} \rightarrow \Re, \sigma \in \Gamma_i\}$, where $\Gamma_i = \{0, 1, 2, ..., \Delta_i\}$, $s_i(t) : [0, \infty) \rightarrow \Gamma_i$ is the switching signal, and Δ_i is the maximal number, which is finite, of the high-order connections for the ith neuron.

It is suggested here to select P_i such that $P_i = \{f_i^0 = \mathbf{w}_i^{0\top} \boldsymbol{\rho}_i^0, f_i^1 = f_i^0 + w_{i,(L_i^0+1)} \rho_{i,(L_i^0+1)}, ..., f_i^\sigma = f_i^{\sigma-1} + w_{i,(L_i^0+\sigma)} \rho_{i,(L_i^0+\sigma)}, \sigma = 2, ..., \Delta_i\}$, where $\mathbf{w}_i^0, \boldsymbol{\rho}_i^0 \in \Re^{L_i^0}$, and L_i^0 is the number of initial high-order connections. Hence, the maximal number of high order connections is $L_i^0 + \Delta_i$.

For simplicity, P_i is rewritten as $P_i = \{f_i^0 = \mathbf{w}_i^{0\top} \boldsymbol{\rho}_i^0, f_i^1 = \mathbf{w}_i^{1\top} \boldsymbol{\rho}_i^1, ..., f_i^\sigma = \mathbf{w}_i^{\sigma\top} \boldsymbol{\rho}_i^\sigma, \sigma = 2, ..., \Delta_i\}$, where $\mathbf{w}_i^1 = [\mathbf{w}_i^{0\top} \; w_{i,(L_i^0+1)}]^\top$ $\boldsymbol{\rho}_i^1 = [\boldsymbol{\rho}_i^{0\top} \; \rho_{i,(L_i^0+1)}]^\top$, $\mathbf{w}_i^\sigma = [\mathbf{w}_i^{(\sigma-1)\top} \; w_{i,(L_i^0+\sigma)}]^\top$ and $\boldsymbol{\rho}_i^\sigma = [\boldsymbol{\rho}_i^{(\sigma-1)\top} \; \rho_{i,(L_i^0+\sigma)}]^\top$ with $\sigma = 2, ..., \Delta_i$.

Let $f_i^0 = \mathbf{w}_i^{0\top} \boldsymbol{\rho}_i^0$ and $f_i^{\Delta_i} = \mathbf{w}_i^\top \boldsymbol{\rho}_i$ be the initial or minimal structure and the maximal structure, respectively, for the ith neuron state. Let the high-order connections, which have

not been connected at $t_\sigma \leq t < t_{\sigma+1}$, be denoted by $\bar{\mathbf{w}}_i^{\sigma\top}\bar{\boldsymbol{\rho}}_i^{\sigma}$ It is noted that for the VSRNN, the time function s_i, defined for the $i-$th neuron, is increasing, with $s_i(0) = 0$.

The function s_i is determined by an external agent, called supervisor; this agent evaluates the VSRNN performance and, depending on this evaluation, it calculates the values of $s_i(t)$ on-line. According to the definitions of the family of functions P_i, and of the switching function s_i, the indexes sequence Σ_i, which is $0, 1, 2, ..., \Delta_i$, is defined off-line. Hence, only the switching time sequence Υ_i is determined, on-line, by the supervisor.

As stated above, the weights are time functions; hence, every w_{ij}, $j = 1, 2, ..., L_i^0$, for the initial structure, is a solution of the differential equation

$$\dot{w}_{ij} = v_{ij}(t),\ w_{ij}(0) \in \Re. \tag{6.3}$$

For $l = L_i^0 + 1, L_i^0 + 2, ..., L_i^0 + k, ..., L_i^0 + \Delta_i$, the weights are given by

$$\dot{w}_{il} = \mu(t - t_k)v_{il}(t),\ w_{il}(0) = 0, \tag{6.4}$$

where $v_{il}(\cdot)$ is a bounded time function defined in Subsection 6.3.1, and $\mu(\cdot)$ is the well-known unit step function. The above equations imply that all the not included weights are zeros, until the respective high-order connections are added. Notice that (6.2) is a continuous switched system, due to the way as we have defined the weights behavior, which are zero until the are connected to the VSRNN.

6.3 Nonlinear System Identification with VSRNN

As done in Chapter 2, consider a nonlinear system of the form (2.3), which can be represented by (2.4) and the optimal weight $\mathbf{w}_i^*$ defined as in (2.5), with zero modelling error term.

Assumption 6.1. The optimal weight vector $\mathbf{w}_i^*$ can be expressed as

$$\mathbf{w}_i^* = [\mathbf{w}_i^{*q\top}\ \bar{\mathbf{w}}_i^{*q\top}]^\top$$

where the entries of $\mathbf{w}_i^{*q}$ can be any finite value and the ones of $\bar{\mathbf{w}}_i^{*q}$ are zeros.

This assumption implies that there could exist a neural structure, simpler than the maximal one, which can approximate arbitrary well the system (2.4). It is worth mentioning that, here, the dimensions of $\mathbf{w}_i^{*q}$ and $\bar{\mathbf{w}}_i^{*q}$ are unknown.

To develop an on-line weights update law, the Series-Parallel model (2.8) is used. The idea is to propose an initial structure for the RHONN and then adapt the neural parameters in such a way that if an error criterion is not satisfied during a period of time, then an extra

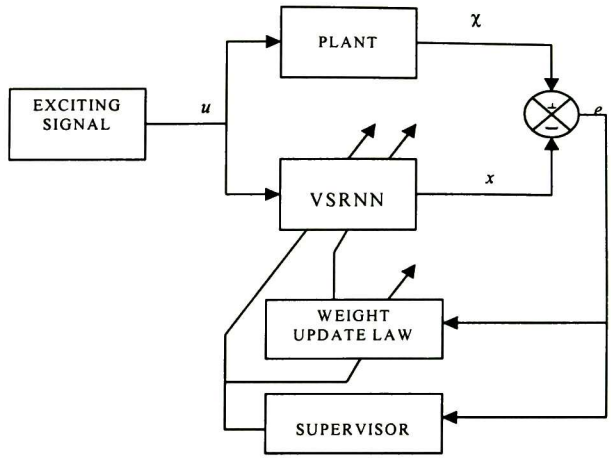


Figure 6.1: Identication block scheme

high-order connection is added. It continues adding different high-order connections, until the error criterion is satisfied or the maximal neural structure is reached. Figure 6.1 shows a block diagram of this scheme, where the supervisor evaluates the VSRNN's performance. For simplicity of notation, the subindex i in the switching function s is dropped.

As in the definition of a VSRNN, one can define $\mathbf{w}_i^* = [\mathbf{w}_i^{*s\top} \quad \bar{\mathbf{w}}_i^{*s\top}]^\top$ where $\mathbf{w}_i^{*s\top}\boldsymbol{\rho}_i^s$ are the added connections to the VSRNN, while $\bar{\mathbf{w}}_i^{*s\top}\bar{\boldsymbol{\rho}}_i^s$ are the not added connections for all $t_s \leq t < t_{s+1}$.

Define the $i-$th identification error as

$$e_i = x_i - \chi_i,$$

and the $i-$th parameter error at $t_s \leq t < t_{s+1}$ as

$$\tilde{\mathbf{w}}_i^s = \mathbf{w}_i^s - \mathbf{w}_i^{*s}$$

Then, from (6.2) and (2.4), one can obtain the error equation

$$\dot{e}_i = -a_i e_i + \tilde{\mathbf{w}}_i^{*s\top}\boldsymbol{\rho}_i^s - \bar{\mathbf{w}}_i^{*s\top}\bar{\boldsymbol{\rho}}_i^s \tag{6.5}$$

6.3.1 On-line Identification

Consider the Lyapunov function candidate

$$V_i = \frac{1}{2}(\gamma e_i^2 + \tilde{\mathbf{w}}_i^\top \tilde{\mathbf{w}}_i), \tag{6.6}$$

where γ is the learning rate. Differentiating (6.6) along the trajectories of (6.5) gives

$$\dot{V}_i = -\gamma a_i e_i^2 + \gamma e_i \tilde{\mathbf{w}}_i^{s\top} \boldsymbol{\rho}_i^s - \gamma e_i \bar{\mathbf{w}}_i^{*s\top} \bar{\boldsymbol{\rho}}_i^s + \tilde{\mathbf{w}}_i^\top \dot{\tilde{\mathbf{w}}}_i \tag{6.7}$$

If one defines the weight adaptive law as [48]

$$\begin{aligned} \dot{\mathbf{w}}_i^s &= -\gamma e_i \boldsymbol{\rho}_i^s \\ \dot{\bar{\mathbf{w}}}_i^s &= 0 \end{aligned} \tag{6.8}$$

then (6.3) and (6.4) are satisfied, and the equation (6.7) becomes

$$\dot{V}_i = -\gamma a_i e_i^2 - \gamma e_i \bar{\mathbf{w}}_i^{*s\top} \bar{\boldsymbol{\rho}}_i^s$$

Considering Assumption 6.1, it is possible to find some intermediate structures, simpler than the maximal structure, such that the term $\bar{\mathbf{w}}_i^{*s\top} \bar{\boldsymbol{\rho}}_i^s \cong 0$. Hence, there exists some finite time, t_q, such that

$$\dot{V}_i = -\gamma a_i e_i^2,$$

for all $t \geq t_q$. This means that, after the qth connection has been added to the VSRNN, one will not consider to add more high-order terms to the network.

Using Lemma 2.1 and Assumption 6.1, it is easy to see that if one further reduces the modelling error term, by adding sufficient connections at a finite time t_q, then with the adaptive law (6.8) one can guarantee that the weights are bounded and the identification error converge to zero after t_q.

6.3.2 Robust On-line Identification

When the modelling error term is not zero, the adaptive law (6.8) does not guarantee either the boundness of the weights or the convergence of the identification error to zero. Therefore, the learning law (6.8) has to be modified in order to avoid the parameters drift problem. For this, the well-known σ-modification scheme [23] is applied to (6.8):

$$\dot{\mathbf{w}}_i^s = -\gamma e_i \boldsymbol{\rho}_i^s - \sigma_i \mathbf{w}_i^s, \tag{6.9}$$

where σ_i is given as

$$\sigma_i = \begin{cases} 0, & \text{if } ||\mathbf{w}_i^s|| \leq M_i \\ \left(\frac{||\mathbf{w}_i^s||}{M_i}\right)^q \sigma_{i_o}, & \text{if } M_i < ||\mathbf{w}_i^s|| \leq 2M_i \\ \sigma_{i_o}, & \text{if } ||\mathbf{w}_i^s|| > 2M_i \end{cases}$$

with integer $q \geq 1$, and σ_{i_0} and M_i are positive constants.

Lemma 6.1 *Let consider the system (2.4) and the VSRNN (6.2), whose parameters are adapted using the law (6.9). Then, $e_i, \mathbf{w}_i^s$ converge into a bounded set.*

Proof. The differential of V_i along the trajectories of (6.5) and (6.9) is given by

$$\dot{V}_i = -\gamma a_i e_i^2 - \sigma_i \tilde{\mathbf{w}}_i^{s\top} \mathbf{w}_i^s - \gamma e_i \bar{\mathbf{w}}_i^{*s\top} \bar{\boldsymbol{\rho}}_i^s.$$

Applying the inequality

$$-\tilde{\mathbf{w}}_i^{s\top} \mathbf{w}_i^s = -(\tilde{\mathbf{w}}_i^{s\top} \tilde{\mathbf{w}}_i^s - \tilde{\mathbf{w}}_i^{s\top} \mathbf{w}_i^{*s}) \leq -\frac{1}{2}|\,||\tilde{\mathbf{w}}_i^s||^2 + \frac{1}{2}|\,||\mathbf{w}_i^{*s}||^2$$

and defining $d_0 = \max_{t\leq 0}(\bar{\mathbf{w}}_i^{*s\top} \bar{\boldsymbol{\rho}}_i^s)$, one obtains

$$\dot{V}_i \leq -\gamma a_i e_i^2 + \frac{\gamma^2 e_i^2}{2} - \frac{1}{2}\sigma_i ||\tilde{\mathbf{w}}_i^s||^2 + \frac{1}{2}\sigma_i ||\mathbf{w}_i^{*s}||^2 + \frac{d_0^2}{2}$$

Define $\alpha = a_i - \frac{1}{2}\gamma$; then,

$$\dot{V}_i \leq -\gamma\alpha e_i^2 - \frac{1}{2}\sigma_i ||\tilde{\mathbf{w}}_i^s||^2 + \frac{1}{2}\sigma_i ||\mathbf{w}_i^{*s}||^2 + \frac{d_0^2}{2}.$$

Substituting e_i from (6.6) in the above inequality yields

$$\dot{V}_i \leq -\alpha V_i + \alpha \tilde{\mathbf{w}}_i^\top \tilde{\mathbf{w}}_i - \frac{1}{2}\sigma_i ||\tilde{\mathbf{w}}_i^s||^2 + \frac{1}{2}\sigma_i ||\mathbf{w}_i^{*s}||^2 + \frac{d_0^2}{2}$$

Considering the worst case, when $||\mathbf{w}_i^s|| > 2M_i$, one can select $\sigma_{io} > 2\alpha$, so that

$$\dot{V}_i \leq -\alpha V_i + \alpha ||\bar{\mathbf{w}}_i^{*s}||^2 + \frac{1}{2}\sigma_{io} ||\mathbf{w}_i^{*s}||^2 + \frac{d_0^2}{2}$$

Therefore, $[e_i \ \tilde{\mathbf{w}}_i^{s\top}]^\top$ converges exponentially on the residual set

$$D_i = \left\{ [e_i \ \tilde{\mathbf{w}}_i^{s\top}]^\top : V_i \leq \frac{1}{\alpha_i}\left(\frac{1}{2}\sigma_{io} ||\mathbf{w}_i^{s*}||^2 + \frac{d_0^2}{2}\right) \right\}$$

and the proof is completed. ■

6.4 VSRNN Supervisor

The aforementioned supervisor criterion, for evaluating the VSRNN performance, is described as follows. Time is divided over the evaluation intervals; each one with a length T called the evaluation period. Define the evaluation function as

$$\dot{E}_i = e_i^2,\ Tk \leq t < T(k+1),\ E_i(Tk) = 0,\ k = 1, 2, ... \tag{6.10}$$

Obviously, $E_i \geq 0$, and it is monotonically increasing for $Tk \leq t < T(k+1)$. If, at the end of each evaluation interval, $E_i(T(k+1)) > c_i$, then the supervisor adds the next high-order connection; otherwise, the neuron keeps its current structure, with c_i being a pre-specified positive bound for E_i.

6.5 Chaos Identification via VSRNN

Recently, controlling and ordering chaos has received increasing attention from various scientific and engineering communities. To control a chaotic system, as for the control of a general nonlinear system, an important requirement usually is to have a good model of the underlying system. For many nonlinear systems, particularly chaotic systems in applications, it is often difficult to obtain an accurate and faithful mathematical model, regarding their physically complex structures and hidden parameters [50]. Therefore, system identification becomes important and even necessary before system control can be discussed.

In order to test the capability and applicability of the proposed scheme, the complex chaotic Chen's system is selected as an example. This chaotic system is described by the following differential equations [55]:

$$\begin{aligned} \dot{\chi}_1 &= a(\chi_2 - \chi_1) \\ \dot{\chi}_2 &= (c-a)\chi_1 - \chi_1\chi_3 + c\chi_2 \\ \dot{\chi}_3 &= \chi_1\chi_2 - b\chi_3\,, \end{aligned}$$

which is chaotic when $a = 35$, $b = 3$, $c = 28$.

The RHONN fixed parameters used in the simulation are the following: $a_1 = 18$, $a_2 = 38$, $a_2 = 28$, $\beta = 0.15$, $I_1 = \{[1\ 0\ 0], [0\ 1\ 0], [3\ 0\ 0], [0\ 3\ 0], [5\ 0\ 0], [0\ 5\ 0], [7\ 0\ 0], [0\ 7\ 0], [9\ 0\ 0], [0\ 9\ 0]\}$, $I_2 = \{[1\ 0\ 0], [0\ 1\ 0], [1\ 0\ 1], [3\ 0\ 0], [0\ 3\ 0], [3\ 0\ 3], [5\ 0\ 0], [0\ 5\ 0], [5\ 0\ 5], [7\ 0\ 7]\}$, $I_3 = \{[0\ 0\ 1], [1\ 1\ 0], [0\ 0\ 3], [3\ 3\ 0], [0\ 0\ 5], [5\ 5\ 0], [0\ 0\ 7], [7\ 7\ 0], [0\ 0\ 9], [9\ 9\ 0]\}$.

The supervisor parameters used are: $c_1 = 0.005$, $c_2 = 0.3$, $c = 0.1$, $T = 10s$.

As can be seen from Figure 6.2, Figure 6.3, and Figure 6.4, the proposed scheme is able to reproduce the Chen's chaotic attractor. Figure 6.5 shows the evaluation functions E_1, E_2, and E_3. The initial and added weights are plotted in Figure 6.6. Finally, in Figure 6.7, the switching signals s_1, s_2 and s_3 are shown.

For further testing of the proposed scheme, the complex chaotic Chua's circuit is considered here. This chaotic system is given by the following differential equations [55]:

$$\begin{aligned} \dot{\chi}_1 &= p(-\chi_1 + \chi_2 - f(\chi_1)) \\ \dot{\chi}_2 &= \chi_1 - \chi_2 + \chi_3 \\ \dot{\chi}_3 &= -q\chi_2, \end{aligned}$$

which is chaotic when $p = 10$, $q = 14.87$ and $f(x) = m_0 x + \frac{1}{2}(m_1 - m_0)(|x+1| - |x-1|)$, with $m_0 = -0.68$ and $m_1 = -1.27$.

The RHONN fixed parameters used in the simulation are the following: $a_1 = 8$, $a_2 = 8$, $a_2 = 8$, $\beta = 0.15$, $I_1 = \{$[1 0 0],[0 1 0],[3 0 0],[0 3 0],[5 0 0],[0 5 0],[7 0 0],[0 7 0],[9 0 0],[0 9 0]$\}$, $I_2 = \{$ [1 0 0],[0 1 0],[0 0 1],[3 0 0],[0 3 0],[0 0 3],[5 0 0],[0 5 0],[0 0 5],[7 0 0]$\}$, $I_3 = \{$[0 1 0],[0 0 1],[0 3 0],[0 0 3],[0 5 0],[0 0 5],[0 7 0],[0 0 7],[0 9 0],[0 0 9]$\}$.

The supervisor parameters used are: $c_1 = 0.0001$, $c_2 = 0.00005$, $c_3 = 0.00005$, $T = 10s$.

As well as in the first reported simulation, Figure 6.8, Figure 6.9 and Figure 6.10 show that the proposed scheme is able to reproduce the Chua chaotic attractor. The evaluation functions E_1, E_2 and E_3 are shown in Figure 6.11, and the x_1 weights and switching signals are plotted in Figure 6.12 and Figure 6.13, respectively.

6.6 Conclusions

In this chapter, the VSRNN has been proposed for nonlinear system identification. This scheme aims to trade off between the identifier performance and the computational complexity. The results are very encouraging, particularly when the scheme is applied to chaos identification. Let us make an analogy between the VSC and the VSRNN scheme, both are divided in two parts; first, sliding mode controller force the system to converge into a desired manifold and the VSRNN supervisor reduces the modelling error term by adding high-order connections, second, once the system is on the sliding manifold it is well-behaved, on the other hand when the modelling error is reduced, such that the error identification tends to zero.

As mentioned in this Chapter the index sequence Σ_i, is defined before the identification process starts. In order to further improve the VSRNN identification performance, the Σ_i sequence could be given by the supervisor, it means that the high-order connections would be

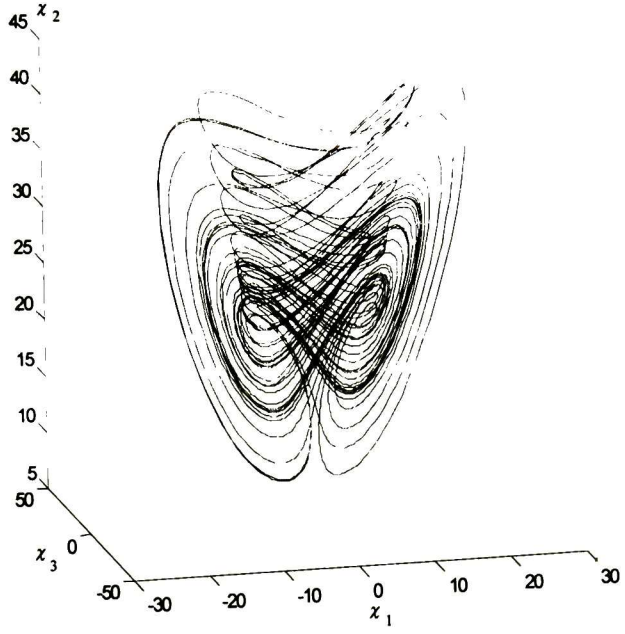


Figure 6.2: Phase portait of Chen's chaotic system

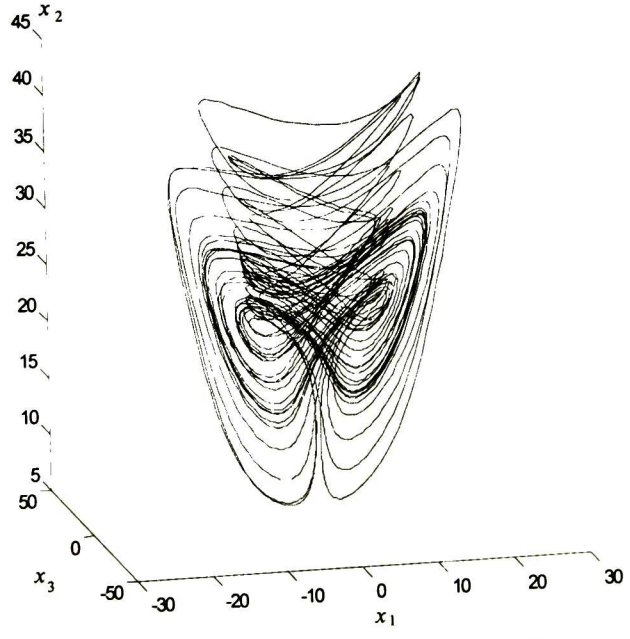


Figure 6.3: Phase portait of the VSRNN

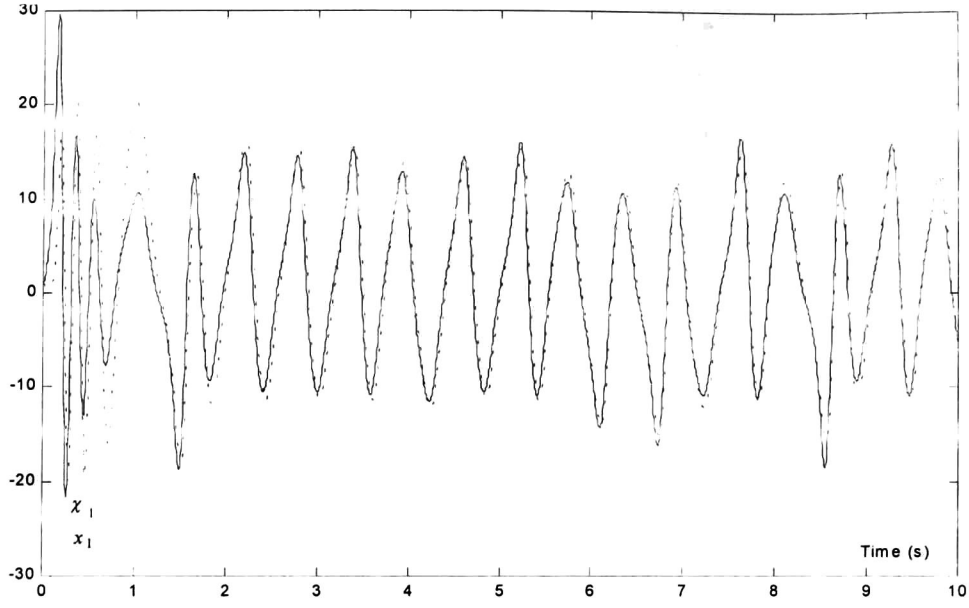


Figure 6.4: χ_1 and x_1 behavior.

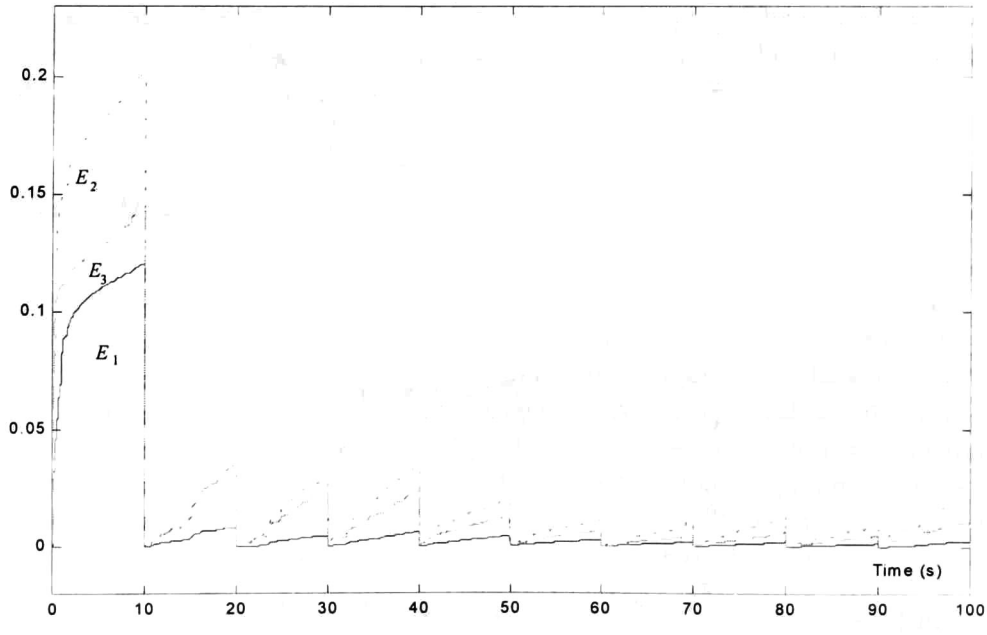


Figure 6.5: Supervisor evaluation functions E_1, E_2 and E_3

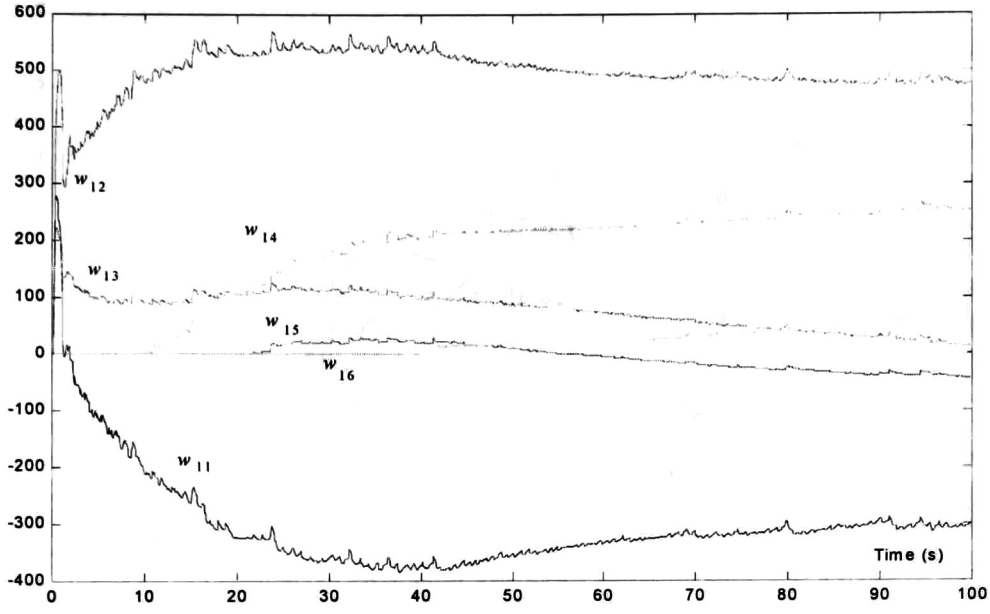


Figure 6.6: Weights of x_1

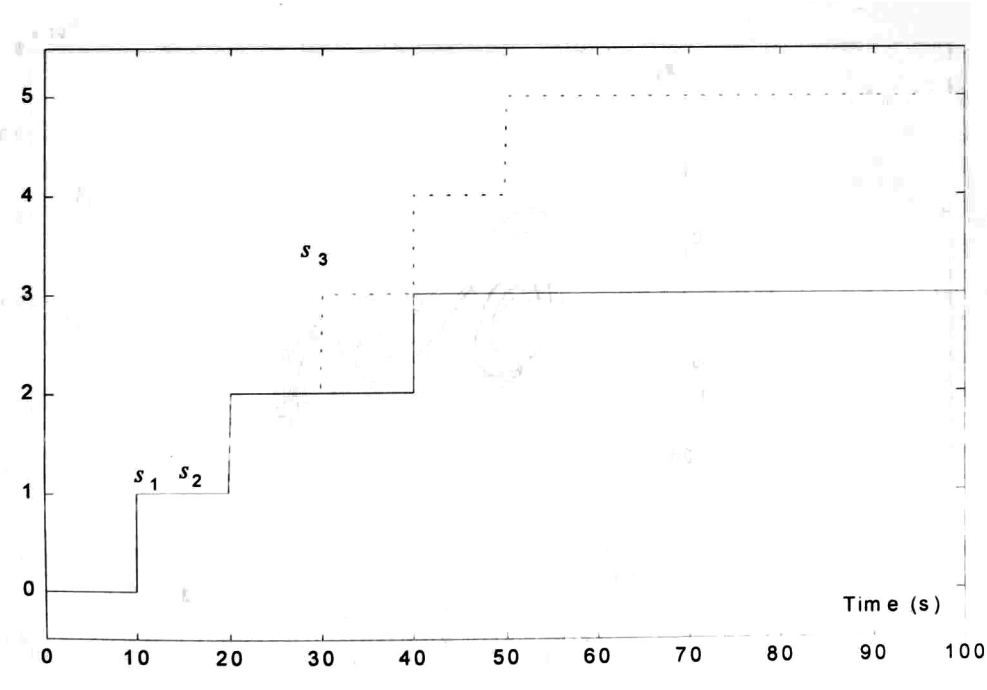


Figure 6.7: Switching signals s_1, s_2 and s_3

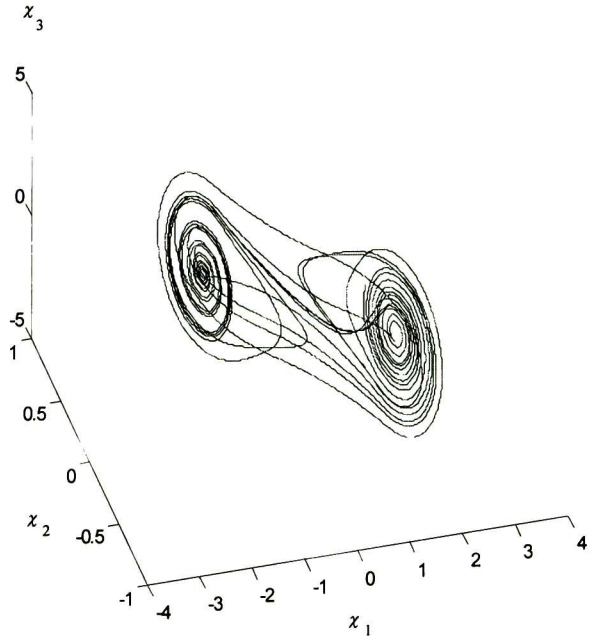


Figure 6.8: Phase portait of Chen's chaotic system

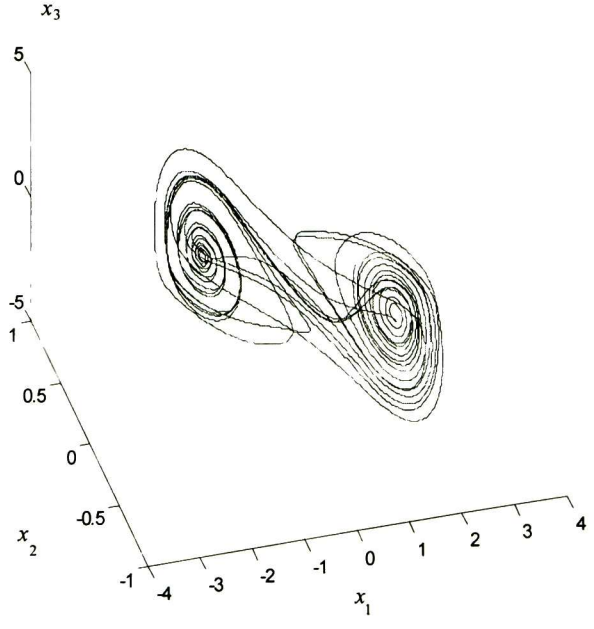


Figure 6.9: Phase portait of the VSRNN

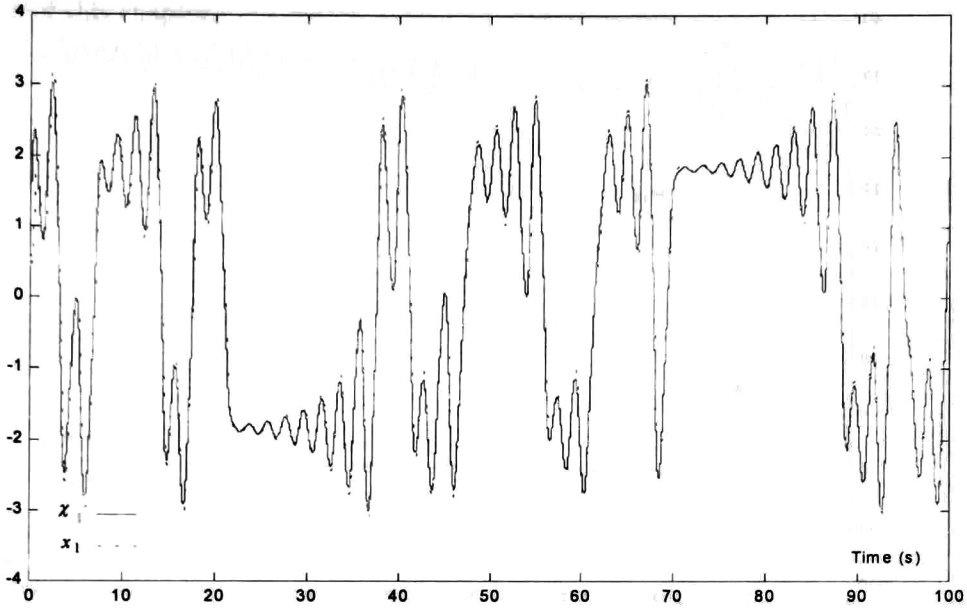


Figure 6.10: χ_1 and x_1 behavior.

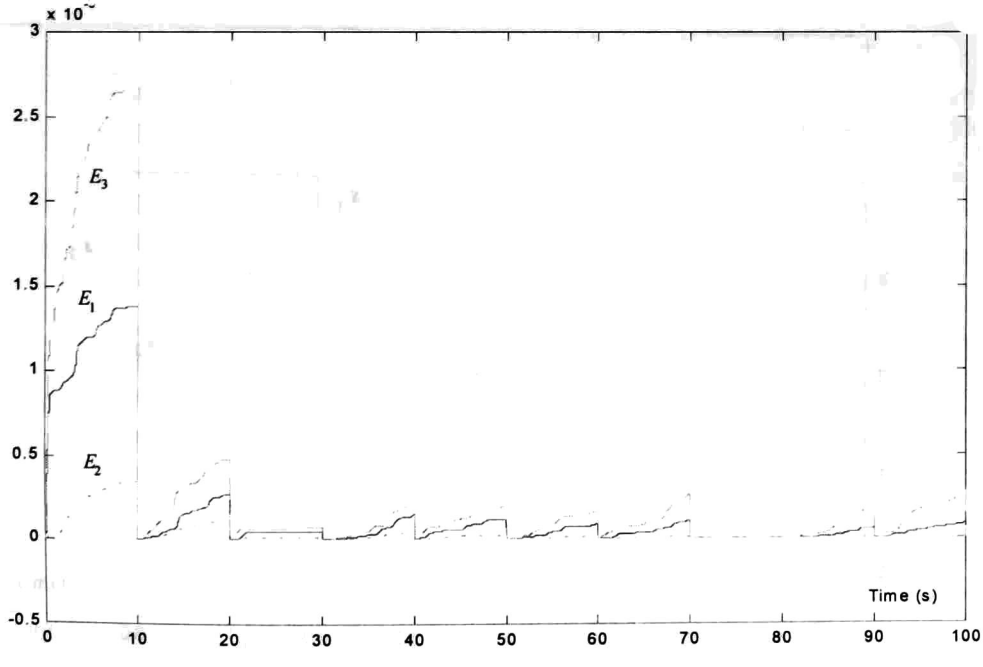


Figure 6.11: Supervisor evaluation functions E_1, E_2 and E_3

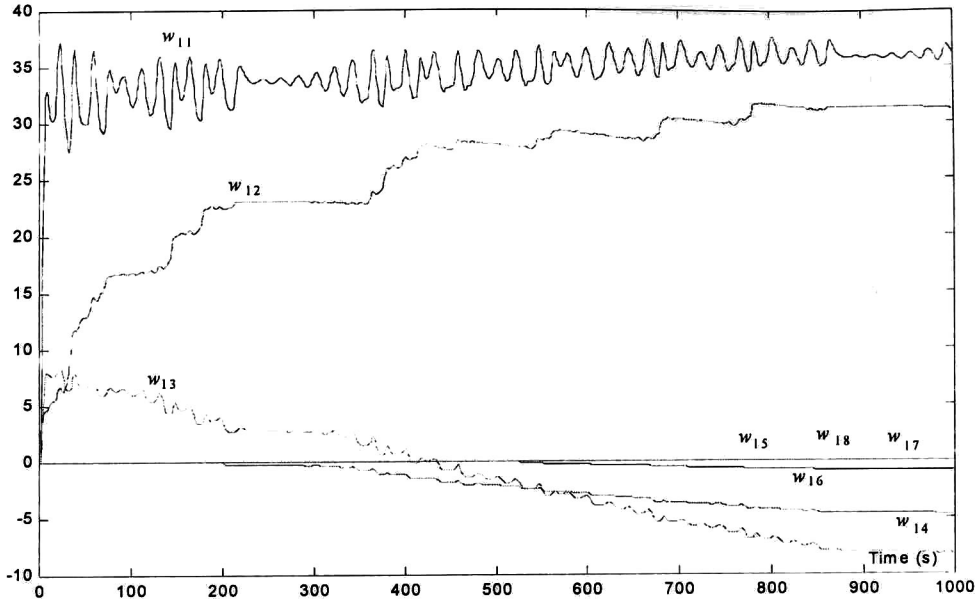


Figure 6.12: Weights of x_1

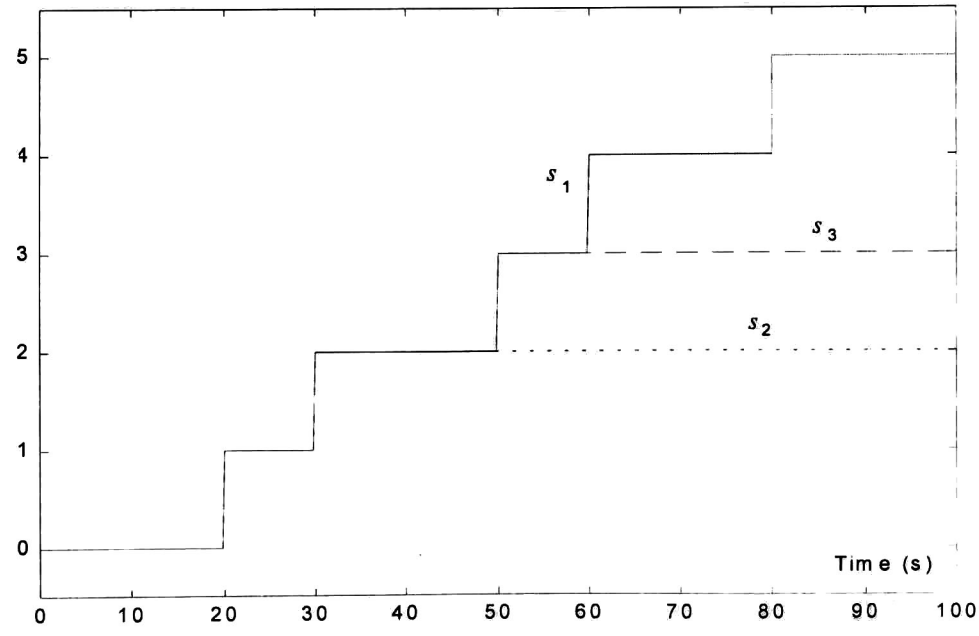


Figure 6.13: Switching signals s_1, s_2 and s_3

connected or disconnected in a arbitrary order. It is worth to mention that [49] and [50] present the results of this chapter.

Chapter 7

Conclusions and Future Work

This thesis, combines VSC and neural networks, in particular the Neural Block Control, which is proposed in Chapter 3 for any block controllable system, resulting in an output tracking controller. Such control strategy consists of proposing a block controllable neural identifier, which updates the adaptive weights, such that, the identification error converge into bounded set (or to zero when the modelling error is zero). For a given output reference the block control algorithm based on the identifier is applied, yielding a discontinuous control law which guarantees the tracking error being ultimate bounded. The overall control system assures that the real plant output tracks the desired output with a small error tracking.

The RHONN are used to design the neural identifiers [28]; the parameter update law was inspired from [48]. To assure the trajectory tracking is needed to guarantee that some identifier parameter do not cross by zero, because the RHONN would lose the controllability property. Such condition is assumed as true for the overall stability analysis. An interesting topic would be to design a parameter adaptive law which assures such weights do not change sign, as well as, the convergence of the identification error to zero, or at least, the convergence into a bounded ball. This problem is left as future research. Other drawback of this scheme is that the controller does not drives the neural outputs error to zero, this obstacle can be eliminated if the relative degree of real plant is equal to 2. Moreover if the modelling error terms are neglected, the overall control strategy guarantees that the real plant output tracks the desired output signal.

This strategy was used to control induction motors, whose model has relative degree equal to 2. In addition to Neural Block Controller, other two controller were designed following the same philosophy; the Singular Sliding Mode (SSM) controller and the Neural Block Control with input constrains. Because the main goal for induction motor control applications is the shaft angular speed regulation, the SSM controller does not control the magnetic flux; this control strategy achieved the best robust performance, but it may generate spurious current

harmonics, making this scheme inviable. For the third control strategy, the Neural Block Control was modified to fit one of the hardest technological constraints, the switching nature of the electrical drives; this control scheme lets to avoid the PWM approach, and yields an actuator model closer to the real one. Additionally this control scheme may reduce the energy dissipated by the motor drive, decreasing heating of the power electronics.

In Chapter 5 the synchronous generator control problem was stated, this system has relative degree 3, so the neural block controller are not able to drive the neural output to the desired one. This disadvantage may makes unreliable this controller, due to the capital importance the output regulation for this kind of systems. Nevertheless, we take advantage from the original model, for which the load angle model has not uncertainties. So an alternative neural block identifier was proposed; based on this identifier the block controller ensures the neural output tracking and therefore the real output tracking when the modelling error is zero. Work is progress to apply this scheme to the decentralized control for multi-machine power systems.

Notice that modelling error term plays an important role on the robustness of these control strategies; if such term is too large, it may makes inviable the proposed control scheme. As mentioned before, the modelling error term can be reduced with a suitable identifier structure, but frequently such structure is not easy to find. In Chapter 6, the VSRNN is proposed to select an accurate structure of the neural identifier before to close the feedback control loop. This identification system is used to establish a model structure, it is useful for tuning and simulation before applying the controller to the real plant. Although the results are very encouraging, particularly when the scheme is applied to chaos identification, VSRNN were not able to find suitable neural structure to identify and control neither induction motors nor synchronous generators. So that, the selection of the identifier structure can be still considered as an open problem for some applications. Nevertheless, the VSRNN identification scheme can be considered as a first step for the structural neural identification, for nonlinear dynamical systems. The VSRNN imposes several constrains; the neurons has to be continuous switched systems, the subsystems are ordered in an increasing complexity way, the index sequence is given off-line, the error criterion is based on the identification error, the high-order connection can only be added. So for future research we propose to relax such constrains, allowing; index sequence given on-line, subsystems not ordered, error criterion based on modelling error and check for not useful high-order connections removing them.

Bibliography

[1] P. Anderson and A. Fouad, *Power System Control and Stability*, Iowa State University Press, 1977.

[2] A. Bazanella, P. Kokotovic and A.S Silva, "A dynamic extension for L_gV controller", *IEEE Transactions on Automatic Control,* vol.44, pp. 588-592, 1999.

[3] F. Blaschke, "The principle of field orientation applied to the new transvector closed-loop control system for rotating field machines", *Siemens Review*, Vol.39, pp. 217-220, 1972.

[4] M. S. Branicky "Multiple Lyapunov Functions and Other Analysis Tool for Switched and Hybrid Systems" *IEEE Transactions on Automatic Control,* vol.43, pp. 475-481, 1998.

[5] K. Bose, *Power Electronics and AC Drives*, Prentice Hall, New Jersey, U.S.A.1986.

[6] G. Buja, D. Casadei and G. Serra "Direct stator flux and torque control of a induction motor: theoretical and experimental results". Proc. IEEE-IECON, pages T50-T64, Aachen Germany, 1998.

[7] Y. Chen. "Decentralized robust control: adaptation for nonlinear uncertainty bound" *Control-Theory Advanced Technology*, vol. 10 pp. 831-846,1994.

[8] N. Cotter. "The Stone-Weiertrass theorem and its application to neural networks", *IEEE Transactions Neural Networks*, vol. 1, no. 4, pp. 290-295. 1990.

[9] M. Depenbrock. "Direct self Control of inverter-fed induction machines" *IEEE Transactions on Power Electronics*, Vol. 3, pp. 420-429, 1998.

[10] B. Drajenovic, "The invariance conditions in variable structure systems" *Automatica*, vol. 5, pp. 287-295, 1969.

[11] R. A. Felix, *Control Neuronal para Motores Eléctricos*, M. Sc. dissertation (in Spanish), CINVESTAV Unidad Guadalajara, México 2000.

[12] R. A. Felix, E. N. Sanchez and A. N. Loukianov "Adaptive Recurrent Neural Control for Electric Power Generators" *American Control Conference.* Denver U.S.A. June 2003.

[13] R. A. Felix, E. N. Sanchez and A. G. Loukianov "Neural Block Control with Input Constraints for Induction Motors" *29 th Conference of the IEEE Industrial Electronics Society.* Roanoke, Virginia, U.S.A, November 2003.

[14] M. Galaz, R. Ortega, A. Bazanella and A. Stankovic, "Excitation Control of Synchronous Generators via Total Energy-Shaping Approach" *American Control Conference*, pp. 817-821, 2001.

[15] M. Ghandhari, G. Anderson, M. Pavella and D. Ernst, "A control strategy for controllable series capacitor in electric power system", *Automatica*, vol.37, pp. 1575-1583, 2001.

[16] D. Gavel and D. D. Siljak, "Decentralized adaptive control: Structural condition for stability," *IEEE Transactions on Automatic Control.* vol 34, pp. 413-426. Apr.1989

[17] S. Grossberg, "Nonlinear networks: principles, mechanisms and architectures". *Neural Networks*, Vol. 1, pp. 17-71, 1988.

[18] M. Gupta and D. H. Rao (Eds.), *Neuro-Control Systems, Theory and Applications*, IEEE Press.

[19] S. Haykin, "*Neural Networks. A comprehensive foundation*" Prentice Hall. New Jersey, U.S.A. 1999.

[20] J. Hopfield, "Neural Networks and physical systems with emergent collective computational abilities", Proccedingsof the National Academy of Science, U.S.A., vol. 79, pp. 2445-2558, 1982.

[21] K. Hunt, G. Irwin, and K. Warwick (Eds.), *Neural Networks Engineering in Dynamic Control Systems,* Springer Verlag, New York, U.S.A. 1995.

[22] P. Iaonnou, "Decentralized adaptive control of interconnected systems," *IEEE Transactions on Automatic. Control,* vol 42, pp. 729-735, Apr. 1986.

[23] P. Ioannou and J. Sun, *Robust Adaptive Control*, Prentice Hall, New Jersey, U.S.A, 1996.

[24] M. D. Ilic and J.W. Chapman, *Decentralized Excitation Control for an Electrical Power Utility*, US patent *#5483147*, January 9, 1996.

[25] A. Isidori, *Nonlinear Control Systems*, Springer Verlag. New York, U.S.A. 1995.

[26] H. Khalil, *Nonlinear Systems*, second edition, Prentice Hall, New Jersey, U.S.A. 1996.

[27] C. A. King, J.W. Chapman and M.D. Ilic, "Feedback linealizing excitation control on full-scale power system model" *IEEE Transactions on Power Systems*, vol. 23, pp.195-211, 2001.

[28] E. Kosmatopoulus, M. M. Polycarpou, M. A. Christodolou and P. A. Ioannou, "High order neural network structures for identification of dynamical systems", *IEEE Transactions on Neural Networks*, vol. 6, pp. 422-431, 1995.

[29] E. Kosmatopoulos, et. al. "Dynamical neural networks that ensure exponential identification error convergence", *Neural Networks*, Vol. 1, No. 2, pp. 299-314. 1997

[30] M. Krstic and H. Deng, *Stabilization of Nonlinear Uncertain Systems*. Springer Verlag. New York, USA.1998.

[31] Z. Krzeminski, "Nonlinear control of induction motors", *10th IFAC World Congress*, pp. 349-354, Munich, Germany. 1987.

[32] W. Leonard, "Microcomputer control of high dynamic performance AC-drives- a survey", *Automatica*, Vol.22, pp. 1-19, 1986.

[33] W. Leonhard *Control of Electrical Drivers*, Springer, New Jersey, U.S.A. 1996.

[34] A. G. Loukianov, "Nonlinear block control with sliding mode", *Automation and Remote Control*, vol.59, No.7, pp. 916-933, 1998.

[35] A. G. Loukianov, J. M. Cañedo, O. Serrano, V. I. Utkin, S. Celikovsky "Adaptive Sliding Mode Block Control for Induction Motor" American Control Conference (ACC), Arlington USA. June 2001.

[36] A. G. Loukianov, E. N. Sanchez and R. A.Felix "Induction Motor VSC Control Using Neural Networks" 15th IFAC World Congress, Barcelona, Spain, July 2002.

[37] A. Luca and G. Ulivi, "Design of exact nonlinear controller for induction motors" *IEEE Trans. Automat. Control*, Vol.34, pp. 1304-1307, 1989.

[38] J. Mancilla and R. Garcia, "On the existence of common Lyapunov triples for ISS and iISS switched systems", in *Proc. IEEE CDC 2000*, Sydney, Australia, December, 2000.

[39] R. Marino, "An example of nonlinear regulator" *IEEE Transactions on Automatic Control*, vol.29, pp. 276-279, 1984.

[40] R. Marino, S. Peresada and P. Tomei, "Adaptive observer-based control of induction motors with unknown rotor resistance" *Int. Journ Adaptive Contr. Signal Proc.*, Vol.10, No.4/5, pp. 345-363, 1996. Piscataway, N.J., 1994.

[41] W. Mielczrsky and A.M. Zajaczkowski, "Nonlinear field voltage control of a synchonous generator using feedback linealization" *Automatica*, vol.30, pp.1625-1630, 1994.

[42] S. Moon, A. Keyhani, and S. Pillutla, "Nonlinear neural network modeling of an induction motor" *IEEE Trans. on Control Systems Technology*, Vol. 7, pp 203-211, 1999.

[43] P. Nicklasson, R. Ortega and G. Espinoza-Pérez, "Passivity-based control of a class of Blondel-Park transformable electric machines", *IEEE Trans. Automat. Control*, Vol.42, No.5, pp. 629-647, 1997.

[44] R. Ortega, N. Barabanov, G. Escobar "Direct Torque Control of Induction Motors: Stability Analysis and Performance Improvement" *IEEE Trans. Aut. Control*, Vol. 46, No. 8, Aug, pp. 1209–1222. 2001

[45] M. Pai, *Energy Function Analysis for Power System Stability*, Klumer Academic Publishers, Boston MA U.S.A. 1989.

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

[47] A. Poznyak, W. Yu, E. Sanchez, and J. Perez, "Nonlinear adaptive trajectory tracking using dynamic neural networks" *IEEE Trans. on Neural Networks*, vol. 10, No 6, pp 1402-1411, Nov. 1999.

[48] G. Rovithakis and M. Christodolou, "Adaptive Control of Unknown Plants Using Dynamical Neural Networks" *IEEE Transactions on Systems, Man and Cybernetics*, vol. 24, pp 400-412. 1994

[49] E. N. Sanchez and R. A. Felix "Nonlinear Identification via Variable Structure Recurrent Neural Networks", *15th IFAC World Congress*, Barcelona, Spain, July 2002.

[50] E. N. Sanchez and R. A. Felix "Chaos Identification using Variable Structure Recurrent Neural Networks" *9th International Conference on Neural Information Processing ICONIP'02*, Singapure. November 2002.

[51] E. N. Sanchez, A. G. Loukianov, R. A. Felix "Stepper Motor Trajectory Tracking via Dynamic Triangular Neural Networks" *15th IEEE International Symposium on Intelligent Control ISIC*. University of Patras Greece. July 2000.

[52] E. N. Sanchez, A. G. Loukianov, R. A. Felix "Recurrent Neural Block Form Control" *Automatica*, No. 39, pp. 1275-1282. 2003

[53] E. N. Sanchez, A. Loukianov, R. A. Felix "Dynamic Triangular Neural Controler For Stepper Motor Trajectory Tracking" *IEEE Transactions on Systems, Man and Cybernetics*, February 2002.

[54] E. N. Sanchez, A. G. Loukianov, R. A. Felix "Stepper Motor Trajectory Tracking Combining Dynamic Triangular Neural Networks and Sliding Modes" *6th International Workshop on Variable Structure Systems*, Gold Coast, Australia. December 2000.

[55] E. N. Sanchez, J. Perez and G. Chen. "Using dynamical networks to generate chaos: An inverse optimal control approach", *International Journal of Bifurcation Chaos*, Vol 11, no. 3, pp 857-863. 2001.

[56] A. Shabanovic and D. Izosimov, "Application of sliding modes to induction motor control" *IEEE Trans.Ind.Applicat.*, Vol. 17, pp. 41-49, 1981.

[57] K. Suykens, L. Vandewalle, and R. de Moor. "*Artificial Neural Networks for Modelling and Control of Nonlinear Systems*" Kluwer Academic Publishers, Boston U.S.A. 1996.

[58] V. I. Utkin, *Sliding Modes in Control and Optimisation*, Springer Verlag, New York, USA, 1992.

[59] V. I. Utkin, "AC drives control problems" *Avtomatica i Telemekhanika*, No. 12, pp. 53-65, 1993.

[60] V. I. Utkin, J. Guldner and J. Shi, *Sliding Modes Control in Electromechanical Systems,* Taylor & Francis, London England.1999.

[61] Y. Wang, D. J. Hill, R. H. Middleton and L. Gao, "Transient stability enhancement and voltage regulation of power system," *IEEE Transactions on Power System*, vol.8, pp.620-627, 1993.

[62] B. Widrow, D. Rumelhart and M. Lehr, "Neural networks: applications in industry, business and science" *Comunications of the ACM*, March, Vol 7, No. 3, 1987.

[63] H. Ye, A. N. Michel and L. Hou "Stability Theory for Hybrid Dynamical Systems" *IEEE Transactions on Automatic Control,* vol.43, pp. 461-474, 1998.

Appendix A

Block Control with Input Constraints for Induction Motors

In this Appendix, the Block Control with input constrains for the induction motor is developed based on the regular model (4.1). Following the sliding mode control strategy, the first step is to design a manifold in the state space, where the tracking errors are asymptotically stable. The block control technique [34] is used to derive such sliding manifold. The next step is to select a discontinuous control law, which guarantees that such manifold is attractive.

The model (4.1) has a quasi NBC (Nonlinear Block Controllable) form, consisting of two blocks:

$$\begin{aligned} \dot{\mathbf{x}}_1 &= \mathbf{f}_1(\mathbf{x}_1) + \mathbf{B}_1(\mathbf{x}_1)\mathbf{x}_2 \\ \dot{\mathbf{x}}_1 &= \mathbf{f}_2(\mathbf{x}) + \mathbf{B}_2\mathbf{u} \end{aligned} \tag{A.1}$$

with $\mathbf{x} = [\mathbf{x}_1, \mathbf{x}_2]^\top$ $\mathbf{x}_1 = [\omega, \psi_\alpha, \psi_\beta]^\top$ $\mathbf{x}_2 = [i_\alpha, i_\beta]^\top$ $\mathbf{u} = [u_\alpha, u_\beta]^\top$

$$\mathbf{f}_1(\mathbf{x}_1) = \begin{bmatrix} -c_0 T_L \\ -c_2\psi_\alpha - n_p\omega\psi_\beta \\ -c_2\psi_\beta + n_p\omega\psi_\alpha \end{bmatrix} \quad \mathbf{B}_1(\mathbf{x}_1) = \begin{bmatrix} -c_1\psi_\beta & c_1\psi_\alpha \\ c_3 & 0 \\ 0 & c_3 \end{bmatrix}$$

$$\mathbf{f}_2(\mathbf{x}) = \begin{bmatrix} c_4\psi_\alpha + c_5 n_p\omega\psi_\beta - c_6 i_\alpha \\ c_4\psi_\beta - c_5 n_p\omega\psi_\alpha - c_6 i_\beta \end{bmatrix} \quad \mathbf{B}_2 = \begin{bmatrix} c_7 & 0 \\ 0 & c_7 \end{bmatrix}$$

This model can be reduced to the NBC-form, and therefore the block control methodology [34] can be applied. ω_r and φ_r are the reference signals for the output variables ω and $\varphi = |\Psi|^2 = \psi_\alpha^2 + \psi_\beta^2$, respectively. The tracking errors for the speed and the flux magnitude are defined as

$$\mathbf{z}_1 = \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} = \begin{bmatrix} \omega - \omega_r \\ \varphi - \varphi_r \end{bmatrix} \tag{A.2}$$

Then, the tracking error dynamics can be expressed as the first block of the NBC-form:

$$\dot{\mathbf{z}}_1 = \bar{\mathbf{f}}_1(\mathbf{x}_1) + \bar{\mathbf{B}}_1(\mathbf{x}_1)\mathbf{x}_2 - \dot{\mathbf{y}}_r \tag{A.3}$$

where $\mathbf{z}_2 = \begin{bmatrix} z_4 \\ z_5 \end{bmatrix}$ is the vector of the new variables, $\bar{\mathbf{f}}_1(\mathbf{x}_1) = \begin{bmatrix} \bar{f}_1(\mathbf{x}_1) = f_1(\mathbf{x}_1) \\ \bar{f}_2(\mathbf{x}_1) = -2c_2\varphi \end{bmatrix}$ $\bar{\mathbf{B}}_1(\mathbf{x}_1) = \begin{bmatrix} -c_1\psi_\beta & c_1\psi_\alpha \\ 2c_3\psi_\alpha & 2c_3\psi_\beta \end{bmatrix}$, $\mathbf{y}_r = \begin{bmatrix} \omega_r \\ \varphi_r \end{bmatrix}$ $\mathbf{K} = \begin{bmatrix} k_1 & 0 \\ 0 & k_2 \end{bmatrix}$ and $\text{rank}(\bar{\mathbf{B}}_1(\mathbf{x}_1)) = 2$.

The desired linearized dynamics for (A.1) is chosen as

$$\dot{\mathbf{z}}_1 = -\mathbf{K}\mathbf{z}_1 + \mathbf{z}_2 \tag{A.4}$$

Solving the equations (A.1) and (A.4) for $\mathbf{z}_2$ results on

$$\mathbf{z}_2 = \bar{\mathbf{f}}_1(\mathbf{x}_1) + \bar{\mathbf{B}}_1(\mathbf{x}_1)\mathbf{x}_2 - \dot{\mathbf{y}}_r + \mathbf{K}\mathbf{z}_1 := \boldsymbol{\alpha}_2(\mathbf{x}_1, \mathbf{x}_2, \mathbf{y}_r, \dot{\mathbf{y}}_r)$$

Then the dynamics for $\mathbf{z}_2$ is

$$\dot{\mathbf{z}}_2 = \bar{\mathbf{f}}_2(\mathbf{x}, \mathbf{y}_r, \dot{\mathbf{y}}_r) + \bar{\mathbf{B}}_2(\mathbf{x}_1)\mathbf{u} \tag{A.5}$$

where $\bar{\mathbf{f}}_2(\mathbf{x}) = \frac{\partial \boldsymbol{\alpha}_2}{\partial \mathbf{x}_2}\dot{\mathbf{x}}_1 + \frac{\partial \boldsymbol{\alpha}_2}{\partial \mathbf{x}_2}\mathbf{f}_2 + \frac{\partial \boldsymbol{\alpha}_2}{\partial \mathbf{y}_r}\dot{\mathbf{y}}_r + \frac{\partial \boldsymbol{\alpha}_2}{\partial \dot{\mathbf{y}}_r}\ddot{\mathbf{y}}_r$, $\bar{\mathbf{B}}_2(\mathbf{x}_2) = \bar{\mathbf{B}}_1(\mathbf{x}_1)\mathbf{B}_2$ and $\text{rank}(\bar{\mathbf{B}}_2(\mathbf{x}^1)) = 2$.

Now, we select the desired sliding manifold as $\mathbf{z}_2 = 0$. The next step is to design a control law which forces the system to reach the desired manifold. This controller must be a logic function which maps from the continuous state $\mathbf{x}$ to the admissible input vectors set U (see Table 4.1), in order to switch the control binary variables, such that, the desired voltage feeds the induction motor.

A.1 Sliding Modes Controller Design

Notice that the term $\bar{\mathbf{B}}_2(\mathbf{x}_1)\mathbf{u}$ describes a $\mathbf{x}_1-$dependent bilinear transformation, which maps from the discrete input space U to the continuous space, or $\bar{\mathbf{B}}_2(\mathbf{x}_1)\mathbf{u} : U \rightarrow \Re^2$ Because the matrix $\bar{\mathbf{B}}_2(\mathbf{x}_1)$ has full rank for all $\psi_\alpha, \psi_\beta \neq 0$, it maps from two different elements of U onto different vectors in $\Re^2$ Now, we rewrite

$$\bar{\mathbf{B}}_2(\mathbf{x}_1) = \mathbf{B}_2'\mathbf{B}_0(\mathbf{x}_1)$$

where $\mathbf{B}_2' = \begin{bmatrix} c_1c_7 & 0 \\ 0 & 2c_3c_7 \end{bmatrix}$ $\mathbf{B}_0(\mathbf{x}_1) := \begin{bmatrix} -\psi_\beta & \psi_\alpha \\ \psi_\alpha & \psi_\beta \end{bmatrix}$

Hence, the mapping $\bar{\mathbf{B}}_2$ can be seen as a composition of two transformations defined by $\mathbf{B}_0(\mathbf{x}_1)$ and $\mathbf{B}_2'$.

In order to have a simple terminology, we establish the following definition:

Definition A.1 *A 2-by-1 vector set covers all the quadrants in $\Re^2$ space, if at least one of them lies on each one of the space quadrants.*

As can be seen in Figure 4.2, U covers all the quadrants of the input space. Let V be the image of the set U under the mapping $\mathbf{B}_0$. By the following lemma states that V also covers all the quadrants.

Lemma A.1 *The angle between any pair of vectors of U is equal to the one between their corresponding images under the mapping B_0.*

Proof. $\mathbf{B}_0$ is a symmetric type matrix, whose columns $\mathbf{b}_{01}$ and $\mathbf{b}_{02}$ are orthogonal and $\mathbf{b}_{01}^\top \mathbf{b}_{01} = \mathbf{b}_{02}^\top \mathbf{b}_{02} = \varphi$. It follows that $\mathbf{B}_0^\top \mathbf{B}_0 = \varphi \mathbf{I}$, hence $\|\mathbf{B}_0\mathbf{u}\| = \sqrt{\mathbf{u}^\top \mathbf{B}_0^\top \mathbf{B}_0 \mathbf{u}} = \sqrt{\varphi}\,\|\mathbf{u}\|$. Let $\mathbf{u}_i$ and $\mathbf{u}_j$ be two different elements in U. By the Euclidean inner product, we have

$$\mathbf{u}_i^\top \mathbf{u}_j = \|\mathbf{u}_i\|\,\|\mathbf{u}_j\| \cos(\theta)$$

where θ is the angle between the vectors. Let $\mathbf{v}_i$,$\mathbf{v}_j \in V$ be the images of $\mathbf{u}_j$ and $\mathbf{u}_j$, respectively, then

$$\mathbf{v}_i^\top \mathbf{v}_j = \|\mathbf{v}_i\|\,\|\mathbf{v}_j\| \cos(\theta') \tag{A.6}$$

with θ' the angle between $\mathbf{v}_i$ and $\mathbf{v}_j$. Now substituting $\mathbf{v} = \mathbf{B}_0\mathbf{u}$ in (A.6) we obtain

$$\mathbf{u}_i^\top \mathbf{B}_0^\top \mathbf{B}_0 \mathbf{u}_j = \varphi \mathbf{u}_i^\top \mathbf{u}_j = \varphi\,\|\mathbf{u}_i\|\,\|\mathbf{u}_j\| \cos(\theta')$$

then $\cos(\theta') = \cos(\theta)$, and the proof is completed. ■

The following lemma states the sliding mode controller viability, using only the available input vectors given by the Table 4.1.

Lemma A.2 *There exists at least one available input vector $\mathbf{u}_k \in U$, such that*

$$\mathbf{sign}(\bar{\mathbf{B}}_2(\mathbf{x}_1)\mathbf{u}_k) = -\mathbf{sign}(\mathbf{z}_2), \textit{for all } z_2 \textit{and } \psi_\alpha, \psi_\beta \neq 0 \tag{A.7}$$

Proof. This lemma statement is equivalent to say that the images of U under $\bar{\mathbf{B}}_2$ covers all the quadrants. From Lemma A.1, it follows that V covers all the quadrants. Now let W be the image of V under $\mathbf{B}_2'$. It is clear that this mapping just changes the axes scale. Hence, it does not move any vector from its original quadrant, hence W covers all the quadrants too. Then, we can conclude that there exists, at least, one image of the elements of U under $\bar{\mathbf{B}}_2$ on each quadrant of $\Re^2$. Hence, there is one or more available input vectors that satisfies (A.7). ■

The controller must select one element of U that satisfies (A.7), and then change the binary variables such that the selected input vector is fed to the stator windings.

A.2 Controller Stability Analysis

At this stage, we analyze the controller stability. Let $\bar{\mathbf{b}}_{21}$ and $\bar{\mathbf{b}}_{22}$ be the rows of $\bar{\mathbf{B}}_2(\mathbf{x}_1)$, then the term $\bar{\mathbf{B}}_2(\mathbf{x}_1)\mathbf{u}_k$ can be expressed as

$$\bar{\mathbf{B}}_2(\mathbf{x}_1)\mathbf{u}_k = \begin{bmatrix} \bar{\mathbf{b}}_{21}\mathbf{u}_k \\ \bar{\mathbf{b}}_{22}\mathbf{u}_k \end{bmatrix} = \begin{bmatrix} sign(\bar{\mathbf{b}}_{21}\mathbf{u}_k)|\bar{\mathbf{b}}_{21}\mathbf{u}_k| \\ sign(\bar{\mathbf{b}}_{22}\mathbf{u}_k)|\bar{\mathbf{b}}_{22}\mathbf{u}_k| \end{bmatrix} = \begin{bmatrix} |\bar{\mathbf{b}}_{21}\mathbf{u}_k| & 0 \\ 0 & |\bar{\mathbf{b}}_{22}\mathbf{u}_k| \end{bmatrix} \begin{bmatrix} sign(\bar{\mathbf{b}}_{21}\mathbf{u}_k) \\ sign(\bar{\mathbf{b}}_{22}\mathbf{u}_k) \end{bmatrix}$$

Using (A.7), the expression (4.19) can be formulated as

$$\bar{\mathbf{B}}_2(\mathbf{x}_1)\mathbf{u}_k = \begin{bmatrix} |\bar{\mathbf{b}}_{21}\mathbf{u}_k| & 0 \\ 0 & |\bar{\mathbf{b}}_{22}\mathbf{u}_k| \end{bmatrix} \mathbf{sign}(\bar{\mathbf{B}}_2(\mathbf{x}_1)\mathbf{u}_k) = - \begin{bmatrix} |\bar{\mathbf{b}}_{21}\mathbf{u}_k| & 0 \\ 0 & |\bar{\mathbf{b}}_{22}\mathbf{u}_k| \end{bmatrix} \mathbf{sign}(\mathbf{z}_1)$$

Then (A.5) can be rewritten as

$$\dot{\mathbf{z}}_2 = \bar{\mathbf{f}}_2(\mathbf{x}, \mathbf{y}_r, \dot{\mathbf{y}}_r) - \mathbf{B}'\mathbf{sign}(\mathbf{z}_2) \tag{A.8}$$

where

$$\mathbf{B}' = \begin{bmatrix} |\bar{\mathbf{b}}_{21}\mathbf{u}_k| & 0 \\ 0 & |\bar{\mathbf{b}}_{22}\mathbf{u}_k| \end{bmatrix}$$

The state (A.8) reaches the sliding manifold $\mathbf{z}_2 = \mathbf{0}$ in a finite time. It can be checked by the Lyapunov function $V(\mathbf{z}_2) = \frac{1}{2}\mathbf{z}_2^\top \mathbf{z}_2$, whose time derivative is definite negative, if the following inequalities about the entries $\bar{\mathbf{f}}_2(\mathbf{x}, \mathbf{y}_r, \dot{\mathbf{y}}_r)$ are satisfied

$$\left|\bar{f}_{21}(\mathbf{x}, \mathbf{y}_r, \dot{\mathbf{y}}_r)\right| < \left|\bar{\mathbf{b}}_{21}\mathbf{u}_k\right| \text{ and } \left|\bar{f}_{22}(\mathbf{x}, \mathbf{y}_r, \dot{\mathbf{y}}_r)\right| < \left|\bar{\mathbf{b}}_{21}\mathbf{u}_k\right|, \quad \forall t \tag{A.9}$$

see [34] and [58] for more details.

The sliding dynamics, on the tracking errors variables z_1 and z_2 (A.2), is governed by the second order linear system

$$\begin{aligned} \dot{z}_1 &= -k_1 z_1 \\ \dot{z}_2 &= -k_2 z_2 \end{aligned}$$

with desired eigenvalues $-k_1$ and $-k_2$.

Note that some of the available input vectors, presented in Table 4.1 may satisfy the condition (A.7); we select the vector which maximizes $\|\bar{\mathbf{B}}_2(\mathbf{x}_1)\mathbf{u}_k\|$, in order to increase the sliding motion stability margin given by (A.9).

Centro de Investigación y de Estudios Avanzados del IPN
Unidad Guadalajara

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

VARIABLE STRUCTURE NEURAL CONTROL

del (la) C.

Ramón Antonio FELIX CUADRAS

el día 24 de Octubre de 2003.

Dr. Bernardino CASTILLO TOLEDO
Investigador Cinvestav 3B
CINVESTAV GDL
Jalisco

Dr. Edgar Nelson SANCHEZ CAMPEROS
Investigador Cinvestav 3B
CINVESTAV GDL
Jalisco

Dr. Alexander GEORGIEVICH LOUKIANOV
Investigador Cinvestav 3B
CINVESTAV GDL
Jalisco

Dr. Juan Manuel RAMÍREZ ARREDONDO
Investigador Cinvestav 3A
CINVESTAV GDL
Jalisco

Dr. Iouri ORLOV
Profesor Investigador
Departamento de Electrónica
y Telecomunicaciones, CICESE
Ensenada B.C.

Dr. Gerardo René ESPINOSA PEREZ
Profesor Investigador
División de Estudios de Posgrado
Facultad de Ingeniería de la UNAM
Mexico D.F.