xx (110526.1)

# CINVESTAV

*Centro de Investigación y de Estudios Avanzados del IPN*
*Unidad Guadalajara de Ingeniería Avanzada*


## *"EQUIVALENTES DINÁMICOS NEURONALES"*



*Tesis presentada por*


**RODRIGO JOEL GARCÍA VALLE**




*Para obtener el grado de*

***Maestro en Ciencias***

*En la especialidad de*

***Ingeniería Eléctrica***

Guadalajara Jal., Junio 2003.

# CINVESTAV

*Centro de Investigación y de Estudios Avanzados del IPN*
*Unidad Guadalajara de Ingeniería Avanzada*


# *"NEURAL DYNAMIC EQUIVALENTS"*

*Thesis submitted by*


**RODRIGO JOEL GARCIA VALLE**


*For the degree of*
***Master in Science***

*In the specialty of*
***Electrical Engineering***

Guadalajara Jal., June 2003.

# Equivalentes Dinámicos Neuronales

## Tesis de Maestría en Ciencias
## Ingeniería Eléctrica


Por:

**Rodrigo Joel García Valle**

Ingeniero Electricista
Escuela Superior de Ingeniería Mecánica y Eléctrica 1997-2001



Becario CONACYT, expediente no. **175213**


Director de Tesis
Dr. Juan Manuel Ramírez Arredondo


CINVESTAV del IPN Unidad Guadalajara, Junio del 2003

# Neural Dynamic Equivalents

**Tesis de Maestría en Ciencias**
**Ingeniería Eléctrica**


Por:

**Rodrigo Joel García Valle**

Ingeniero Electricista
Escuela Superior de Ingeniería Mecánica y Eléctrica 1997-2001



Becario CONACYT, expediente no. **175213**


Director de Tesis
Dr. Juan Manuel Ramírez Arredondo


CINVESTAV del IPN Unidad Guadalajara, Junio del 2003

# Resumen

Se presenta una técnica para construir equivalentes dinámicos a través de la preservación de los modos electromecánicos. El sistema externo se reduce a unos pocos generadores ficticios y sus parámetros son estimados bajo dos procesos de optimización diferentes, Algoritmos Genéticos (AG) y Levenberg-Marquardt. Esta técnica es capaz de preservar la estructura modal asociada al sistema de estudio. Se realizan simulaciones en el tiempo para comparar las señales obtenidas, tanto del sistema completo como del sistema reducido.

Al mismo tiempo, la aplicación de técnicas de Inteligencia Artificial como los son las Redes Neuronales Artificiales (RNA) son empleadas para la resolver la difícil tarea de la construcción de Equivalentes Dinámicos. El objetivo principal es crear Equivalentes Dinámicos Robustos apoyados por una red neuronal artificial capaz de poder reproducir el voltaje en los nodos frontera. Esta novedosa proposición para desarrollar Equivalentes Dinámicos Robustos evita el problema del cálculo de los parámetros de los generadores equivalentes así como también la linealización del sistema de potencia. Para construir Equivalentes Dinámicos Robustos se tomaron en cuenta diferentes condiciones operativas y de igual forma se consideran sistemas de control como lo son los estabilizadores de sistemas de potencia (PSS).

# Abstract

A methodology to construct dynamic equivalents through matching modes associated to internal generators is presented. This technique is based on the classical methods where the electric power system must be linearized for its analysis. The formulation is posed as an optimization problem with an objective function based on eigenvalues' second order sensitivities. The external system is reduced to a few fictitious generators, whose parameters are to be estimated by two different optimization procedures, Genetic Algorithms (GA) and Levenberg-Marquardt. This method is able to preserve the modal structure associated with the study system.

Furthermore, the application of Artificial Intelligence techniques such as Artificial Neural Networks (ANN) is employed to solve the hard task of constructing Dynamic Equivalents. The main objective is to create Robust Dynamic Equivalents assisted by an ANN able to reproduce the complex voltage at frontier buses. This novel proposition to develop Robust Dynamic Equivalents evades the problem to compute the parameters for the equivalents generators as well as avoid the linearization of the power system. To structure Robust Dynamic Equivalents different operation conditions are taken account in addition to consider control systems for instance power system stabilizer (PSS).

**To my parents,**

**Jaime and Lidia**

# Acknowledgments

I would like to express for his great gratitude and valuable friendship to my adviser, Prof. Juan M. Ramirez Arredondo, for his many helpful suggestions, much encouragement and assistance all through the course of this work. Likewise, I want to be thankful for his patience and confidence given to me during the performing of this work.

I would like to give thanks to those people who has helped and contributed in many different ways with practical experience, technical and non-technical suggestions.

I am thankful primarily to the National Council of Science and Technology (CONACYT) for its support and sponsorship.

I would like to express my deep admiration, appreciation and respects to my parents Jaime and Lidia for their advice, support, understanding, patience, love and much affection throughout all my life and in those difficult moments. As well, to my sister Cynthia for her great affection, love and her wonderful friendship.

Finally, I feel like to express to my partner and girl-friend Citlalli for her great trust, tolerance, friendliness, companionship, huge affection and love during all this time.

# Tables and Figures

**TABLES**



**FIGURES**

# Acronyms

| | |
|---|---|
| AC | Alternate Current. |
| ADALINE | Adaptive Linear Neuron. |
| AI | Artificial Intelligence. |
| ANN | Artificial Neural Networks. |
| ART | Adaptive Resonance Theory. |
| ARX | Auto Regressive Extra input. |
| BP | Back-Propagation. |
| CMAC | Cerebellar Model Articulation Control. |
| DC | Direct Current. |
| Emfs | Electro-motive forces. |
| FACTS | Flexible Alternate Current Transmission Systems. |
| GA | Genetic Algorithms. |
| Hz | Hertz. |
| kV | Kilovolts. |
| LMS | Least Mean Squared. |
| LVQ | Learning Vector Quantizer. |
| MIMO | Multiple inputs-Multiple Outputs. |
| MISO | Multiple inputs-Single Output. |
| MLP | Multi-Layer Perceptron. |
| MSE | Mean Squared Error. |
| MVA | Mega Volt-Ampere. |
| PRBS | Pseudo Random Binary Signal. |
| PSS | Power System Stabilizer. |
| REI | Radial, Equivalent and Independent. |
| RMS | Root Mean Squared. |
| SA | Simulated Annealing. |
| SMIB | Single Machine Infinite Bus. |
| SOFM | Self-Organising Feature Maps. |
| UPFC | Unified Power Flow Controller. |
| ZMIID | Zero Mean, Independent, Identically Distributed Disturbance. |

# Contents

# Chapter 1

# INTRODUCTION

*Everything should be made as simple as possible . . . but not simpler!*
*A. Einstein*

## 1.1 Electricity: Historical Background.

Electric power has performed a worthy role to the progress and technological advances of the human being over the last century. Edison Electric of New York founded the first *direct current* (DC) electric power station in 1881. It was a limited system due to it could deliver power only a short distance from generation. Thanks to the development of the transformer a few years later the first *alternate-current* (AC) system was installed in Massachusetts in 1886 by Westinghouse. That was the beginning of the AC transmission systems and this fact marked a pause for the following years to the construction of the polyphase systems by Nicola Tesla. Through the years all countries have had the necessity to structure an electric power grid that will be proficient to satisfy all their requirements.

Every one electric power system around the world differs from size, frequency range, generation, transmission and load capacity mainly. However, all of them have the same structure and they are comprised by three principal elements which are the generating units, transmission lines and distribution systems; nowadays, control equipment has a valuable function and they also constitute a part of whichever electric power system.

*Generating units* are the most important elements in every electric power systems. Electric power is generated using synchronous machines, which are driven through turbines that can be steam, hydraulic, wind, diesel, nuclear or internal combustion. The generator voltages are regularly in the range of 11-35 kV. Usually, the generating stations are far away from consumers' centres. *Transmission lines* are the connections among distribution systems and generating units. They interconnect all key power generating units and most important load centres in the system. The transmission system is the backbone of any electric power system and operates at the highest voltage levels (commonly, 230 kV-400 kV and above). *Distribution system* is the last constituent of the electric power system. The primary distribution voltage is typically between 4.0 kV and 34.5 kV. Small industrial consumers are provided by primary feeders at these voltage levels; residential and commercial consumers are supplied by a secondary distribution feeder where the voltage level is 120/240 volts. Figure 1.1 shows the basic elements of a power system.

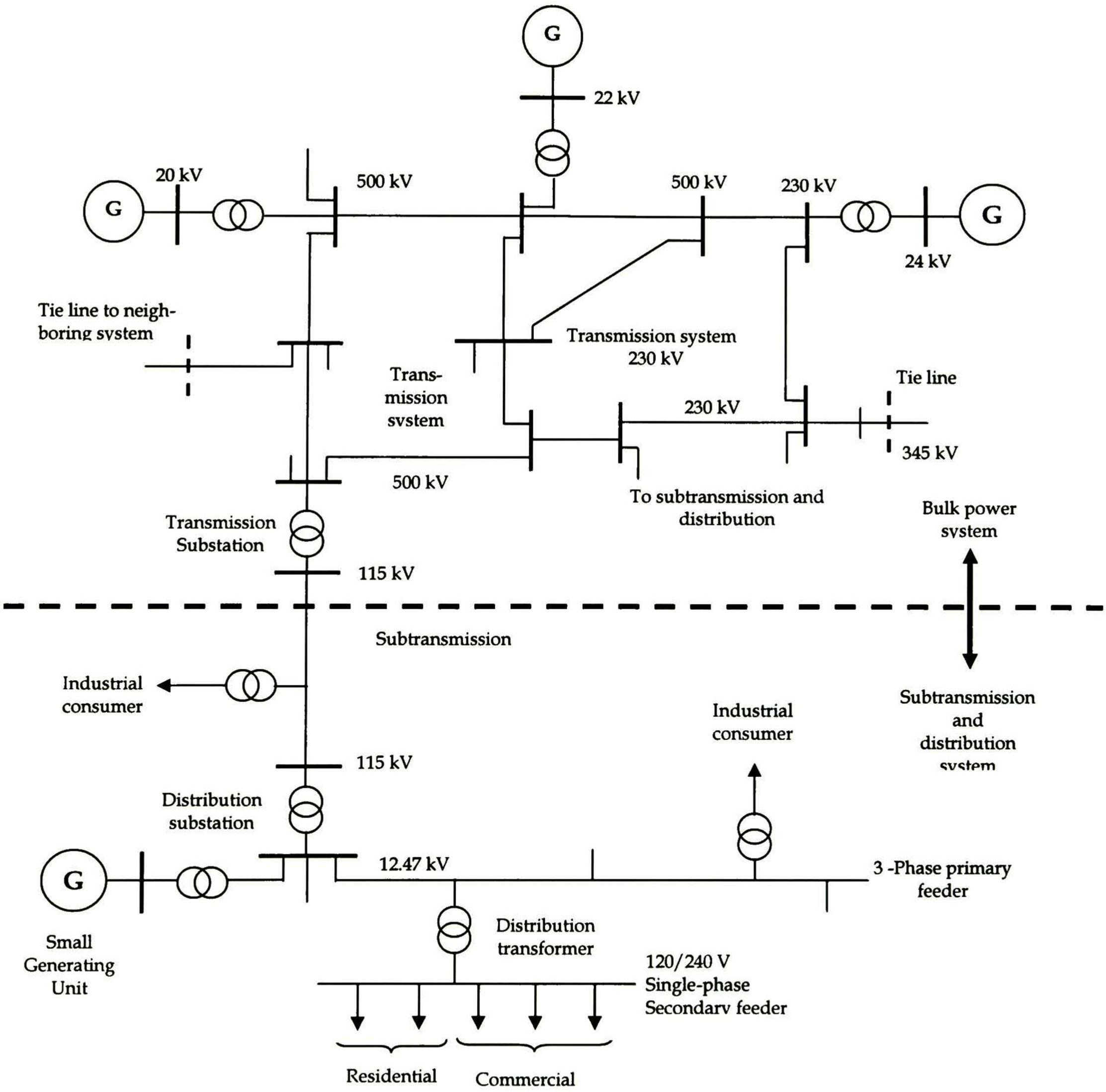



Figure 1.1 Basic elements of a power system.

Constantly, the task to get a successful drive of any power system has been considered as a bulky labour for electrical engineers; however they commit themselves to provide a good service to all the consumers, which it must be reliable and stable. The customers must be supplied by electric energy with frequency and voltage as constant parameters. Hence, these parameters have to be into certain tolerance limits such that the clientele's devices can function in a proper manner; i.e. the voltage drop must not exceed ± 10% from its standard boundary, and the frequency must not surpass 1 Hertz from its nominal value.

## 1.2 Motivation.

The electric power system analysis have always been characterized to be a hard duty to face due to all the issues that they represent, bearing in mind the complex topic that they signify. This challenging task has been confronted by different ways and by many researchers worldwide. There are too many notable, successful and important results achieved in this area but, in spite of everything there continue a vast quantity of problems that are hardly difficult to solve employing recent advances in numerical analysis and decision support systems. Commonly, these troubles are summarised in the following manner [1].

- Inappropriate model of the real world.
- Complexity and size of the problems which prohibit computation time.
- Solution methods employed by the human are not capable of being expressed in an algorithm or mathematical form. They usually involve many rules of thumb.
- The operator decisions are based on fuzzy linguistics descriptions.
- Analysis of security related with voltage or angle is based on human experience judgment and experience.

Owing to all the preceding drawbacks and the great computational innovations that have been evolved for the human well-being, important mechanisms to develop modern techniques to solve these kinds of problems have come up. Thus, for the last years researchers have done many efforts to develop new approaches based on Artificial Intelligence in order to improve on speed, accuracy, efficiency, and ability to handle stressed/ill-conditioned systems. The main branches in Artificial Intelligence are Fuzzy Logic, Artificial Neural Networks, and Expert Systems.

These novel techniques but, in particular the Artificial Neural Networks, have been tested to solve many problems obtaining outstanding solutions. Artificial Neural Networks are able to overcome many tasks such as classification, clustering, pattern recognition and forecasting among many other applications corresponding to different areas. From a dynamic power system standpoint, the application of the AI methods has presented great results [1-9].

In this work the efficiency and feasibility of the Artificial Neural Networks (ANN) to predict events and/or signal is proved to obtain Dynamic Equivalents. Owing to their great many potential applications in power systems planning and operation, dynamic equivalents have attracted much research attention worldwide over the last decades. Here-to-fore, the motivation to develop accurate, low-order dynamic equivalent models has been aimed at reducing the very considerable computing times associated with large-scale transient stability studies, multi-machine power systems. Several methods have been published to advance this research issue but problems remain, particularly in the area of robustness; in other words they have limitations such as the machine model order, many of them do not include static excitation system, power system stabilizers (PSS) or merely the tested system do not include flexible alternate current transmission systems (FACTS) devices, and nowadays

almost the whole electric grids around the world comprise with one of these devices, so then, they take a very important role to bear in mind. Above and beyond these restrictions, all these works have been solved by classical techniques. Thus, these are the main motivations to construct a Dynamic Equivalent that overtake the limitations that others can not do. Moreover, with the advent of market forces in the electricity supply industry, and the ensuing confidential status given to all utility data, network information exchange between neighbouring utilities may be in the form of reduced equivalent circuits. Hence, it becomes essential to develop a new generation of dynamic equivalents that are robust and have self-learning capabilities.

**1.3 Thesis's Structure.**

This work is divided in three stages mainly. In the first one, chapter 2, is presented the power system stability problem including its principal characteristics to can understand this issue and afterwards, a large vision about Dynamic Equivalents is described.

In chapter 3, the Artificial Neural Networks (ANN) are explained. Firstly, the relation among the human brain and the ANN is described and a brief classification of them is also depicted. Then, a mathematical model that represents this approach is described. The main optimisation techniques used for neural networks and their training algorithms are illustrated. Finally, several application examples are depicted standing out the example related to electric power systems.

In the last stage, chapter 4, the most important proposition is made. In this chapter, the application of Artificial Neural Networks to construct Dynamic Equivalents is described. It is presented in a detailed manner how to foresee the complex voltage to developed Dynamic Equivalents supported by an ANN. The obtained results show the feasibility, confidence and robustness of the proposed methodology.

**REFERENCES.**

1. James A. Momoh and Mohamed E. El-Hawary, *Electric Systems, Dynamic, and Stability with Artificial Intelligence Applications.* Marcel Dekker Inc., 2000.
2. Djukanovic M, Sobajic DJ, Pao YH., *Artificial Neural Network Based Identification of Dynamic Equivalents*. Electric Power System Research, 1992; 24: pp. 39-48.
3. Ramírez B. Rogelio, *Evaluación de la Estabilidad Transitoria en Sistemas Eléctricos de Potencia Mediante Redes de Neuronas Artificiales*, PhD Thesis, FIME, Universidad Autónoma de Nuevo León, December, 1994.
4. Wehenkel L, Jacquemart Y., *Use of Kohonen Feature Maps for the Analysis of Voltage Security Related Electrical Distance*. Proceedings of ICANN'95, International Conference of Artificial Neural Networks, October 1995: pp. 8.3.1-8.3.7.

5. Chen Dingguo, *Nonlinear Neural Control with Power Systems Applications,* PhD Thesis, Oregon State University, September, 1998.
6. Wilfert Hans-Helmut, Voigtländer Knut, Erlich Istvan, *Dynamic Coherency Identification of Generators Using Self-Organising Feature Maps*. Elsevier Control Engineering Practice 2001 (9); pp. 769-775
7. Matsubara T, Nakamura K, Fujita H, Sone M., *Transient Stability Criterion Using Neural Networks.* IEE Japan, 2002.
8. Chaturvedi D.K., Malik O. P. and Kalra P. K., *Power System Stabilizer Using A Generalized Neural Network,* NAPS 2002, Arizona State University: pp. 320-327.
9. Tomonobu Senjyu, Shotaro Yamane, et. al, *Improvement of Multi-Machine Power System Stability with Variable Series Capacitor Using On-Line Learning Neural Network,* Electrical Power and Energy Systems 2003; 25: pp.403-409.

facilities, loss of generation, or loss of a large load. The system response to such disturbances involves large excursions of generator rotor angles, power flows, bus voltages, and other system variables. If the resulting angular separation between two machines in the system remains within certain bounds, the system maintains synchronism; if loss of synchronism because of transient stability occurs, it will usually be evident within 2 or 3 seconds from the initial disturbance.

Thus transient stability is a highly non-linear, highly dimensional phenomenon that involves large disturbances and topological changes [1,2].

### 2.1.1 Power Oscillations.

Two types of *synchronizing oscillations* are common in all interconnected AC power systems. The first is associated with a single generator (or a plant of identical generators) acting against the system. The second is more complex and involves many generators; one area of the power system oscillating against generators in other areas of the power system. *Local* or *plant modes* of oscillations have natural frequencies of about 1 to 2 Hz. *Inter-area* modes of oscillation have lower natural frequencies on the order of 0.1 to 0.7 Hz. In small systems, inter-area oscillations generally have higher natural frequencies than those of larger systems[2].

The total number of modes of synchronizing oscillations is equal to one less than the number of interconnected generators. In a system having thousands of generators, there are thousands of oscillating modes. All of these must decay following the system disturbance. If any mode increases in amplitude, the system's operators would have to take action to prevent either a local or a system-wide collapse.

Power systems must be designed to be stable under a range of system loads and operating conditions. Generally, if the operation of the system is constrained, those constraints should be due to the thermal operating limits of the transmission system or loss of synchronism (transient instability) and not by oscillatory instability[3].

To determine the nature of system oscillations, analysis of the following system characteristics is required:

- Frequency and *damping* of the system's synchronizing oscillation.
- Pattern of generators that take part in each mode of oscillation.

Generators that are able to have a controlling effect on the oscillations must be identified, and tools must be provided to allow an efficient and robust design of oscillation damping controls[2].

On the other hand, for small signals other two types of studies can be discussed, *angle instability* and *voltage instability*, which are associated to local modes and inter-area modes.

If the system has an insufficient synchronizing and a damping torque it refers to as angle instability; then if the electric network does not manage sufficient reactive power to support the load, a voltage collapse could occur and that could lead to a voltage instability.

### 2.1.2 Swing Equation.

Bearing in mind the fact that the problem of stability derive from synchronous machines, it is necessary to involve the solution of the *swing equation* for each machine of the system to obtain their *rotor's angle* as a function of time. This is a differential equation governing the motion of machines.

$$M\frac{d^2\delta}{dt^2} = P_a \qquad (2.1)$$
$$P_a = P_m - P_e$$

where

$\delta$ is the displacement rotor angle according to a reference.

$M$ is the inertia constant of machine.

$P_a$ is the accelerating power.

$P_m$ is the mechanical power.

$P_e$ is the electrical power.

The most viable and common technique to solve the swing equation is the *point-by-point* solution. This solution is basically a numerical technique where the accelerating power $P_a$ is assumed constant during a short period of time $\Delta t$, chosen for numerical integration, so that we can easily get the rotor speed $\omega$ and the rotor angle $\delta$ by integrating the swing equation; once to get the rotor speed and twice to obtain the rotor angle. This is only valid for the particular period of time under study.

Using this method we need to keep in mind some considerations [1,13]:

i. The mechanical power input remains constant during the complete study period
ii. The machine is represented by constant voltage after transient reactance
iii. Damping is ignored
iv. Constant flux linkages in each axis
v. No transient saliency exists, this means $X_d^{'} = X_q^{'}$

### 2.1.3 Single Machine and the Equal-Area Criterion.

To identify if the system is stable after a disturbance it is necessary to solve the swing equation. The system is unstable if the angle of a machine or between any two machines tends to increase without limit. By the same way, stipulating that the system is under disturbance effects, if the angle reach a maximum value and decrease afterwards, the system is stable. There is a simple and direct method for determining the stability of the system, and it is not necessary any solution of the swing equation. This

method is known as *equal-area method*. There are some assumptions for applying this method, which basically are:

a. Constant mechanical power.
b. Classical machine model.

Take $\delta$ into the swing equation, (which expresses the motion or swing of the rotor of the machine), as shown in Fig. 2.1; in an unstable system, $\delta$ increases indefinitely with time and the machine loses synchronism. In a stable condition, $\delta$ undergoes oscillations, which eventually disappear because of damping[4]. From Fig. 2.1, it is clear that, for a stable system , $d\delta/dt = 0$ must be satisfied at some time.

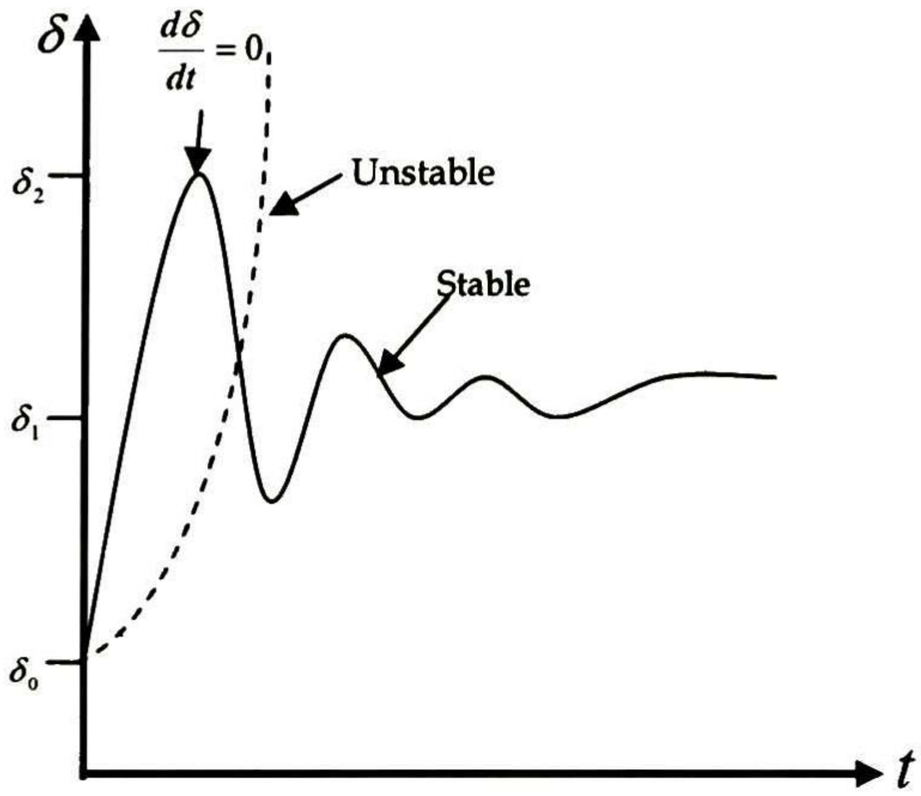



Fig. 2.1 Stable and Unstable system.

Therefore the stability is checked by monitoring the rotor speed deviation $\frac{d\delta}{dt}$, which must be zero at some moment, this means that:

$$\int_{\delta_0}^{\delta} P_a d\delta = 0 \tag{2.2}$$

This condition requires that, for stability, the area under the graph of accelerating power $P_a$ versus $\delta$ must be zero for some value of $\delta$; considering that $M$ is constant and the damping is slight, the positive (or accelerating) area under the graph must be identical to the negative (or decelerating) area [4]. This is recognized as the equal-area method for stability. To get a better idea of this method we may refer to Fig. 2.2.

Point *a*, corresponding to $\delta_0$, is the initial steady-state operating point. At this position, the input power to the machine, $P_{i0}$, is equal to the developed power $P_{e0}$. When a sudden increase of the input power occurs to $P_i$, the accelerating power, $P_a$, becomes positive and the rotor moves towards point *b*. It is assumed that the machine is connected to a large power system and there is also a *constant field current* which maintains the *internal voltage* $|E_g|$ constant. Thus, the rotor accelerates and the power

angle begins to increase, at point $b$, $P_i = P_e$ and also $\delta = \delta_0$ [4]. At this moment, $P_a$ is negative and $\delta$ finally reaches a maximum value $\delta_2$ or point $c$ and then swings backwards $b$. Hence, the rotor establishes to the point $b$, which is the last steady-state stable operating point, as shown in Fig. 2.2.

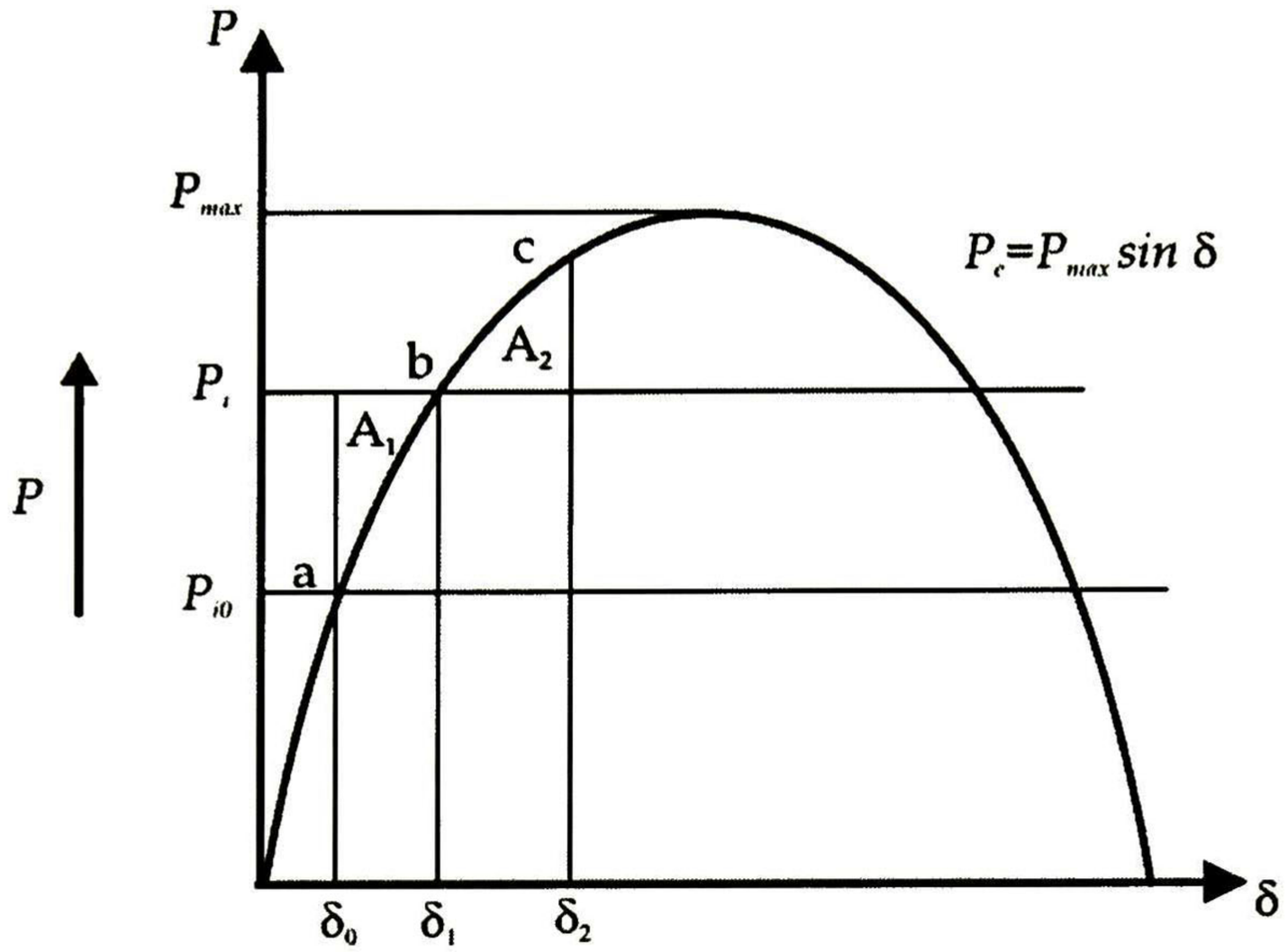



Fig. 2.2.Power angle characteristic.

For stability the equal-area method requires,

$$\text{Area } A_1 = \text{Area } A_2$$

that is,

$$A = A_1 - A_2 = 0$$

or, from equation 2.2, we have,

$$\int_{\delta_0}^{\delta_1} (P_i - P_{\max} \sin\delta) d\delta = \int_{\delta_1}^{\delta_2} (P_{\max} \sin\delta - P_i) d\delta \qquad (2.3)$$

## 2.2 Load Models.

The importance of *load modelling* in power systems studies, such as transient stability studies is well known. Stable operation of a power system depends on the ability to continuosly match the electrical output of generating units to the electrical load on the system [13]. However, the major problem in the evaluation of power system dynamic performance is not represented by the complexity in load model, the main problem is posed by the difficulty in obtaining data.

For stability studies load representation is very common to be denoted by a load bus which represents incandescent and flourescent lamps, heaters, air conditioner, ovens, refrigerators, motors, arc furnaces and so on. Induction motors constitute a major portion of the system load. The precise representation of load is difficult to estimate. Eventhough the precise representation of load is known, there are many

other factors like time (hour, day, season), weather conditions and the state of the economy that prevent the load modelling in power systems. Hence, load representation in stability studies is based on a considerable amount of simplifications.

The real and reactive power of load could be represented by a mathematical model. The load models can be divided into two categories:

a) *Static loads.*
b) *Dynamic loads.*

### 2.2.1 Static Load Model.

A static load model expresses the characteristics of the load at any instant or time as algebraic functions of the bus voltage magnitude and frecuency at that moment [14]. There are two different ways of static load representation.

I. *Polynomial representation.*

In this case, typically both active and reactive power loads are represented by quadratic polynomials given by

$$P = P_0\left[a_0\left(\frac{V}{V_0}\right)^2 + a_1\left(\frac{V}{V_0}\right) + a_2\right]$$
$$Q = Q_0\left[b_0\left(\frac{V}{V_0}\right)^2 + b_1\left(\frac{V}{V_0}\right) + b_2\right] \qquad (2.4)$$

where $V_0$, $P_0$ and $Q_0$ are normally taken as initial operating conditions. This representation is also recognized as *ZIP model,* as it is constituted of a *constant impedance* (*Z*), *constant current* (*I*) and *constant power* (*P*) components. The coefficients $a_{0,1,2}$ and $b_{0,1,2}$ are fractions of the constant power, constant current and constant impedance components in the active power loads. A constraint is imposed:

$$a_0 + a_1 + a_2 = 1$$
$$b_0 + b_1 + b_2 = 1$$

This load representation in not appropriate for cases involving large voltage variations.

II. *Exponential representation.*

This load model has a voltage dependence of load characteristics and it has been represented by,

$$P = P_0\left(\frac{V}{V_0}\right)^{k_{PV}}$$
$$Q = Q_0\left(\frac{V}{V_0}\right)^{k_{QV}} \qquad (2.5)$$

The parameters of this model are the exponents $a$ and $b$. Through these exponents the model represents constant power, constant current or constant impedance characteristics, respectively.
For constant power model, the voltage is invariant and allows loads be represented with a stiff voltage characteristic $k_{PV} \approx k_{QV} \approx 0$. This model is frequently used in *load-flow studies*, but it is not recommended for other analysis, for instance transient stability analysis, or where there are present severe voltage drops. The $k_{PV}$ and $k_{QV}$ constants represent the *voltage sensitivities*, which are frequently expressed in pu with respect to the given operating point.
The constant current model gives a load demand that varies linearly with voltage $k_{PV} \approx 1$. This is a practical representation of the real power demand as a mixture of resistance and motor devices.
Finally, for constant impedance model, the load power changes proportionally to the voltage squared $k_{PV} \approx k_{QV} \approx 2$. This model represents lighting loads but it does not model stiff loads satisfactorily.

### 2.2.2 Dynamic Load Model.

Generally, the *dynamic load modelling* is associated to the study of systems where there are large concentrations of motors. Many studies as inter-area oscillations, voltage stability and *long-term stablity* require dynamic loads to be modelled. Mainly, dynamic load modelling could be represented as an *induction motor*.
The best way to get an induction motor model is to take into account only the dynamics of the rotor inertia described by

$$\frac{d\omega_m}{dt} = \frac{1}{2H}\left[T_E(S) - T_M(\omega_m)\right] \tag{2.6}$$

where $\omega_m$ is the per unit motor speed; $H$ is the inertia constant and the per unit mechanical torque $T_M$ is a function of $\omega_m$ as

$$T_M = T_{M0}(A\omega_m^2 + B\omega_m + C) \tag{2.7}$$

where $A$, $B$, $C$ are defined as constants. The per unit electrical torque $T_E$ is a function of the *motor slip* $S$ and is computed from the steady state equivalent circuit shown in Fig. 2.3. Also, if *rotor flux transients* are to be included the model may be modified.

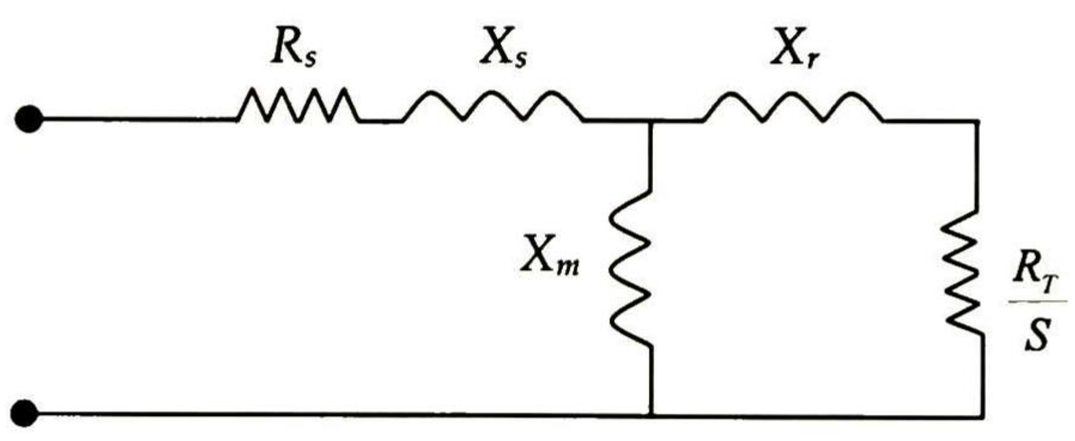



Fig. 2.3 Steady state equivalent circuit of an induction motor.

## 2.3 Synchronous Machine Models.

*Synchronous generators* are the most important and principal sources of electric energy at any power system. The power system stability problem is mainly related keeping to interconnected synchronous machines in synchronism. However, power system dynamic problems are basically those of the synchronous machines. There are many kinds of power system dynamics problems, like high – or low frequency oscillations, and large or small system disturbances. Owing to these problems, several synchronous machine models were developed. Each model is given a number that gives you an idea of the number of *differential equations* that are required to describe the model. The larger of the number means the model complexity, in addition the time required to solve the differential equations depends on the model complexity. It is assumed that all quantities are expressed in per unit [1, 4, 13].

*Model 6* – $(\dot{E}_d'', \dot{E}_q'', \dot{E}_d', \dot{E}_q', \omega, \delta)$

In this model the generator is represented by the *subtransient emfs* (electro-motive forces) $E_d''$ and $E_q''$ behind the subtransient reactances $X_d''$ and $X_q''$. The differential equations that describe this model are given by

$$T_{d0}'' \dot{E}_d'' = E_q' - E_q'' + I_d\left(X_d' - X_d'\right) \tag{2.8}$$

$$T_{q0}'' \dot{E}_d'' = E_d' - E_d'' - I_q\left(X_q' - X_q''\right) \tag{2.9}$$

$$T_{d0}' \dot{E}_q' = E_f - E_q' + I_d\left(X_d - X_d'\right) \tag{2.10}$$

$$T_{q0}' \dot{E}_d' = \quad - E_d' - I_q\left(X_q - X_q'\right) \tag{2.11}$$

$$\dot{\omega} = \frac{1}{M}\left(P_m - P_e - D\omega\right) \tag{2.12}$$

$$\dot{\delta} = \omega - \omega_s \tag{2.13}$$

As the first's two differential equations include the influence of the *damper windings*, the damping coefficient in the swing equation needs only to quantify the mechanical damping because of windage and friction; as this is frequently small, it could be neglected $(D \approx 0)$.

*Model 5* – $(\dot{E}_d'', \dot{E}_q'', \dot{E}_q', \omega, \delta)$

In this model the effect of the *rotor body eddy-currents* in the $q$-axis are neglected, then $X_q' = X_q$ and $E_d' = 0$. This is the classical 5 winding model with *armature transformer emfs* neglected. The equations for this model are

$$T_{d0}'' \dot{E}_d'' = E_q' - E_q'' + I_d\left(X_d' - X_d''\right) \tag{2.8}$$

$$T_{q0}'' \dot{E}_d'' = E_d' - E_d'' - I_q\left(X_q' - X_q'\right) \tag{2.9}$$

$$T_{d0}' \dot{E}_q' = E_f - E_q + I_d\left(X_d - X_d'\right) \tag{2.10}$$

$$\dot{\omega} = \frac{1}{M}\left(P_m - P_e - D\omega\right) \tag{2.12}$$

$$\dot{\delta} = \omega - \omega_s \tag{2.13}$$

This model has two equivalent rotor windings and time constant $\left(T_{d0}^{''}, T_{d0}^{'}\right)$ on the *d*-axis and three armature reactances $X_d^{''}, X_d^{'}, X_d$. In the *q*-axis there is one equivalent rotor winding, with a time constant $\left(T_{q0}^{''}\right)$, and two armature reactances $\left(X_q^{''}, X_q\right)$.

*Model 4* – $(\dot{E}_d^{'}, \dot{E}_q^{'}, \omega, \delta)$

In this model the damper winding effects of Model 6 are neglected, so then equations (2.8) and (2.9) must be removed to get the model. Now, the generator is represented by the *transient emfs* $E_q^{'}$ and $E_d^{'}$ behind the transient reactances $X_d$ and $X_q^{'}$. This synchronous generator model is usually considered to be satisfactorily precise to analyse electromechanical dynamics. The principal disadvantage of this model is that the equivalent damping coefficient that appears in the swing equation cannot be calculated exactly.

*Model 3* – $(\dot{E}_q^{'}, \omega, \delta)$

This model is almost as *Model 4* except that the *d*-axis transient emf $E_d^{'}$ is assumed to remain constant. Thus the equations for this model are given by

$$T_{d0}^{''}\dot{E}_d^{''} = E_q^{'} - E_q^{''} + I_d\left(X_d^{'} - X_d^{''}\right) \tag{2.8}$$

$$\dot{\omega} = \frac{1}{M}\left(P_m - P_e - D\omega\right) \tag{2.12}$$

$$\dot{\delta} = \omega - \omega_s \tag{2.13}$$

*Model 2* – $(\omega, \delta)$

This is the well-known classical synchronous generator model. This model assumes that both the *d*-axis armature current $I_d$ and the internal emf $E_f$ that represents the excitation voltage do not fluctuate during the transient state. At this model, the generator is represented by the swing equation and a constant emf $E^{'}$ behind the transient reactance $X_d^{'}$. The equations for this model are

$$\dot{\omega} = \frac{1}{M}\left(P_m - P_e - D\omega\right) \tag{2.12}$$

$$\dot{\delta} = \omega - \omega_s \tag{2.13}$$

This model is traditionally used in power system analysis and it also can be used for evaluating generator behaviour during the first rotor swing.

## 2.4 Dynamic Equivalents.

As power systems are being increasingly interconnected and due to their dimension, for stability studies it is impossible or not efficient to represent the entire system in detail. Simple equivalents that model the transient behavior of distant generators in response to system changes are desirable [5].

For purpose of analysis, and to get a better sight to put up a reduced equivalent system, the power system network is divided into two parts, which are:

a) *Internal or study system,* is that subsystem where disturbances are to be applied and where the response of machines is to be observed.
b) *External system,* is that area where detailed information on the system response is not required, therefore it is desirable to represent the external system by equivalents.

In order to solve the dimensionality problem - one of the main troubles to solve transient stability in power systems, which involve the data and the time to solve the system-, is suitable to put away the external system and develop the transient stability study only for the internal system. In general, the coupling among these systems cannot be omitted, because of the union between them, which is strong enough; hence, reliable results will not be obtained.
A good solution for the dimension of the problem can be to find a technique to reduce the size of the external system. This reduction needs to be in the way that the impact of the behavior between the external system and the study system must be the same. On the reduced system will be possible to simulate a disturbance as if it was the external system, and reliable results should be obtained. A reduced-order model for the external system that carries out these objectives is called a *dynamic equivalent system*. The purpose of equivalents is to reduce computer storage, time requirements, and the corresponding overwhelm analysis [6]. That is, dynamic or electromechanical equivalents are reduced-order differential equation models, useful in operating systems due to the limitations of memory capability, computation time and also the time to prepare information and analysis results.

Talking about dynamic equivalents it is necessary to define some important terms:

- *Dynamic aggregation.*
- *Coherent groups.*

The order of the differential system equations representing the dynamic part of the system can be reduced by grouping units that are in parallel on the same bus, and replacing them by an equivalent generating unit [7, 15], this is what it is called *dynamic aggregation.*
In contrast, *coherent groups* of generating units for a given perturbation are defined as a group of generators oscillating with the same *angular speed,* and terminal voltages in a constant complex ratio.

The main approaches used to derive power system dynamic equivalent for transient stability studies may be classified in six groups [7]:

1. Empirically based simplifications.

2. Methods based on linearization and modal analysis.
3. Methods based on coherency.
4. Modal-coherent equivalents.
5. Identification dynamic equivalents.
6. Dynamic equivalents obtained using singular perturbation theory.

The most used and important techniques are: the *modal approach,* the *coherency approach* and the *estimation approach* [8-12, 15-18]. A brief description of these techniques will be useful to understand them.

The modal approach studies the modes of the linearized system, in order to eliminate the less significant ones for the contingency of concern [11, 16, 18].

The coherency methods are based on the existence of coherent groups of machines during transients, their identifications and their aggregation. Coherency analysis is based on the generators' time of response following a disturbance, like a network fault [5, 10, 15, 17].

Modal-coherent equivalent can be derived preserving not only the coherent groups of the original system model, but also the modes of group-to-group oscillations. It is constructed only once for a given utility and can be used in the transient stability study of any disturbance [11, 16].

System identification refers to the determination of the essential characteristics of a dynamic system when observing the response of system variables to random system inputs, either natural or intentional [8, 10, 12]. The major advantage of this technique is that information of the external system is not required.

## 2.5 Method Based Upon Coherency.

The theory of coherency is originally applied to generator buses as a basis for reducing the number of buses in the power grid. In this case, two buses are defined as coherent if the complex voltages relations are constant over the time. In practice, it is common to say that two buses are coherent only by examining their voltages angles; as a consequence two buses are considered coherent if their angular difference is constant to a certain tolerance over the period of simulation [15, 17].

The overall procedure for forming coherency-based dynamic equivalents can be divided into five fundamental steps:

i. Definition of the study area.
ii. Identification of coherent generators.
iii. Generator buses reduction.
iv. Load buses reduction.
v. Dynamic aggregation of generating unit models.

The next stride is to include a pithy clarification of each one of the above steps.

### 2.5.1 Definition of the Study Area.

The study area defines that area of the network that will be retained in detail. Regularly, this area will be specified by a list of buses that will not be eliminated and a list of generating units that will not be aggregated [5, 11, 15, 17]. The study area does not necessarily need to be contiguous. An essential point here is to consider the load model. If loads are modelled as constant impedances, the requirements of the equivalency procedure are minimal. Hence, it is just necessary to retain only those buses that are involved in switching operation and pass up aggregating generating units which are close to the fault. On the other hand, if non-linear components will be taken in load models a traditional technique to define the study area is needed; it is necessary to keep an area which surrounds the fault to avoid reducing non-linear load buses that experience large voltage changes. Another requisite to take into account non-linear loads is that the power system or network will be subdivided into sub-areas, which are harshly coherent, and the tie lines among the sub-areas will be retained.

### 2.5.2 Identification of Coherent Generators.

Two generator buses are defined as coherent if their angular difference is closely invariable within a predefined tolerance over a certain period. It is necessary to consider the coherency of both internal and terminal generator buses, because the first one forms the basis for the network reduction. The coherent groups of generators can be defined by a specific fault occurring inside an area but it is essential to describe the fault type. In a few words, a procedure for identifying coherency for a single fault will be described.

First, it is necessary to form a simple model of the power system that uses the following assumptions [5, 8, 9, 15, 17]:

- The coherent groups of generators are completely independent from the size of the disturbance. For that reason, coherency could be determined by taking into consideration a linearized system model.
- The amount of detail in the generating unit is independent of the coherent groups. Thus, a classical synchronous machine model will be supposed and the excitation and turbine-governor system will be ignored.
- To reproduce the fault effect on the power system, the mechanical output will be pulsed to attain the same accelerating power, like if a fault would have existed.

Presently, a description of each assumption will be done. For the first assumption, the coherency behavior of the generators does not change radically as the fault clearing time increased. The next assumption is based on the fact that even though the amount of detail in the generating unit models has a considerable consequence ahead on the swing curves, specially the damping, it does not affect the most essential characteristics, such as natural frequencies and mode shapes. Finally, the third assumption accepts that the generator accelerating powers are roughly constant during faults with typical clearing time.

The mechanical equation for the motion of a synchronous machine must be linearized with the real power equations decoupled from the reactive power ones and the resultant equation must be:

$$\begin{bmatrix} \Delta PG \\ \Delta PL \end{bmatrix} = \begin{bmatrix} \partial PG/\partial\delta & \partial PG/\partial\theta \\ \partial PL/\partial\delta & \partial PL/\partial\theta \end{bmatrix} \begin{bmatrix} \Delta\delta \\ \Delta\theta \end{bmatrix} \tag{2.14}$$

where

$\Delta$ represents incremental variables.

*PG* are the real power injections in internal generator buses.

*PL* are the real power injections in load buses.

$\delta$ are the angles in generator internal buses.

$\theta$ are the angles in load buses.

An integration technique will be used to get a time domain solution of the linearized swing equation. The trapezoidal approach is perfect for this purpose, since it is not necessary a precise solution and while the method is numerically stable, large step sizes can be used. Afterwards, a clustering algorithm is used to estimate the swing curves that are obtained for the linear recreation and, in that way, the coherent groups will be determined. The coherency analysis may well be applied to several sets of swing curves for different faults to establish an equivalent that is suitable to a range of disturbances.

### 2.5.3 Coherency Measures.

The final report on EPRI project RP904, entitled *"Coherency Based Dynamic Equivalents For Transient Stability Studies"* [17], reported that some *coherency measures* which are based in the internal voltage angle deviation, has presented high-quality results for dynamic equivalents. These two kinds of measures are: the *max-min measure* and the *RMS measure*.

The RMS coherency measure is a criterion used for determining whether a unit should be added to an existing group. If the approximate swing curves are clustered, the criterion for coherency is:

$$\left|\Delta\delta_l(t) - \Delta\delta_k(t)\right| < \varepsilon \tag{2.15}$$

for all the samples of time *t*.

where:

*l* is the index for generator being clustered.

*k* is the index for reference generator for the group under consideration.

The RMS coherency measure evaluated over an infinite interval can be analytically related to generator inertias, synchronizing torque coefficients of equivalent lines connecting internal generator buses and the statistics of the system disturbance for step input disturbances [16].

The max-min coherency measure is defined as:

$$r_0 = \left| \frac{\max_i x_i(0) - \min_i x_i(0)}{\max_i x_i(0)} \right|$$
$$r_1 = \left| \frac{\max_i \dot{x}_i(0) - \min_i \dot{x}_i(0)}{\max_i \dot{x}_i(0)} \right| \qquad (2.16)$$
$$r_2 = \frac{\min_{i,j} Y_{ij}}{\max_i Y_{i\infty}}$$
$$r_3 = \frac{\max_i M_i - \min_i M_i}{\max_i M_i}$$

where

$x_i = \theta_i - \theta_i^s$

$\theta_i$ is the torque angle of machine $i$.

$\theta_i^s$ is the steady state value of $\theta_i$.

$Y_{ij}$ is the magnitude of transfer admittance between machines $i$ and $j$.

$r_o$ and $r_1$ measure the degree of differences between the initial conditions.

$r_2$ measures the degree of coupling among the machines relative to the coupling to the infinite bus.

$r_3$ defines the similarity of the machines inertias.

There are differences (advantages/disadvantages) between the RMS and the max-min measures; some of them will be described below.

- The coherent groups determined by the max-min coherency measure are dependent on a disturbance location.
- The max-min coherency measure produces better results than the RMS coherency measure, when they are compared for a short period of observation.
- The max-min coherency measure produces better results if the purpose is to get a dynamic equivalent for a specific disturbance.
- The RMS coherency measure has been algebraically related to the parameters of the system model and the statistics of the modal disturbance.
- The RMS coherency measure can determine the coherent groups without the necessity of simulation, as it is required for the max-min coherency measure.
- The RMS coherency measure reflexes in a superior manner the total dynamic of the external system than the max-min coherency measure.

### 2.5.4 The Zero Mean, Independent, Identically Distributed Disturbance (ZMIID).

This type of disturbance is commonly applied for constructing modal-coherent dynamic equivalents. This kind of disturbance has several advantages than other types of disturbances. The disturbance is

fundamentally the mean value of a step on the mechanical input power; it has other characteristics like the disturbance independency from all the other generator buses; it also is identically distributed, therefore its name of ZMIID.

The main purpose of this technique is to establish a clear relationship between modal and coherency analysis with the aim of developing an equivalent, that combines both approaches [11, 16]. This equivalent is constructed by applying an RMS coherency measure, evaluated over an infinite interval to identify coherent groups in face of the Zero Mean, Independent, Identically Distributed (ZMIID) step input disturbance.

### 2.5.5 Modal-Coherent Equivalents.

The modal approach consists of creating a reduced linear model, by typical eigenvalue techniques, which maintains a selected number of oscillation modes. The crucial part is the preliminary choice of the rule of mode elimination, which should take into account the disturbances. The most important advantages of the modal approach is that the equivalent need to be computed only once for any given unit commitment, network configuration and load flow [18] and after that it may be used to study many different system disturbances.

The most important advantage of this technique is that the appearance of the equivalent formed is a reduced set of equivalent generators and lines, which can be used completely in transient stability programs. As a result, the combination of these two approaches has some properties that are [18]:

1. The system eigenvalues will not be required in order to construct the equivalent.
2. The eigenvalues of the equivalent will closely approximate the system eigenvalues retained by the modal equivalent based on the same disturbance and RMS coherency measure.
3. The equivalent will be useful for studying any disturbance that might occur outside the coherent groups aggregated to form the equivalent.
4. Power system component structure is retained and the equivalent can be used with existing transient stability studies.

Hence, the modal-coherent technique has properties to construct attractive and robust equivalents instead of modal or coherent equivalents.

## 2.6 Generator Buses Reduction.

A physical interpretation of the generator bus reduction will be described. To do this physical interpretation the simple network given in Fig. 2.4a and complementary Figs. will be useful to exemplify the course of action and a simple consideration will be taken. The generator terminal buses 1, 2 and 3 have been recognized as coherent and they are going to be substituted by a sole equivalent bus 4 [15, 17].

Step 1.

It is necessary to define the voltage $\bar{V}_t$ in the equivalent network; it could be done by selecting the voltage of an individual bus or an average voltage of the group; this is with the aim of minimizing the variation in the internal machine voltages which takes place in consequence of the machines being transferred to the equivalent bus. All buses are linked via an ideal transformer with a complex turn ratio to the equivalent bus. The turn ratio can be calculated as: $\bar{a}_k = \bar{V}_k / \bar{V}_t$, where: $\bar{V}_k$ is the voltage on bus *k*.

Under coherent circumstances, the ratio $\bar{a}_k$ is invariable for each bus in the group and there is not circulating power flow by any of the phase shifters. The purpose is that the phase shifters will not affect on the voltages and currents of the network.

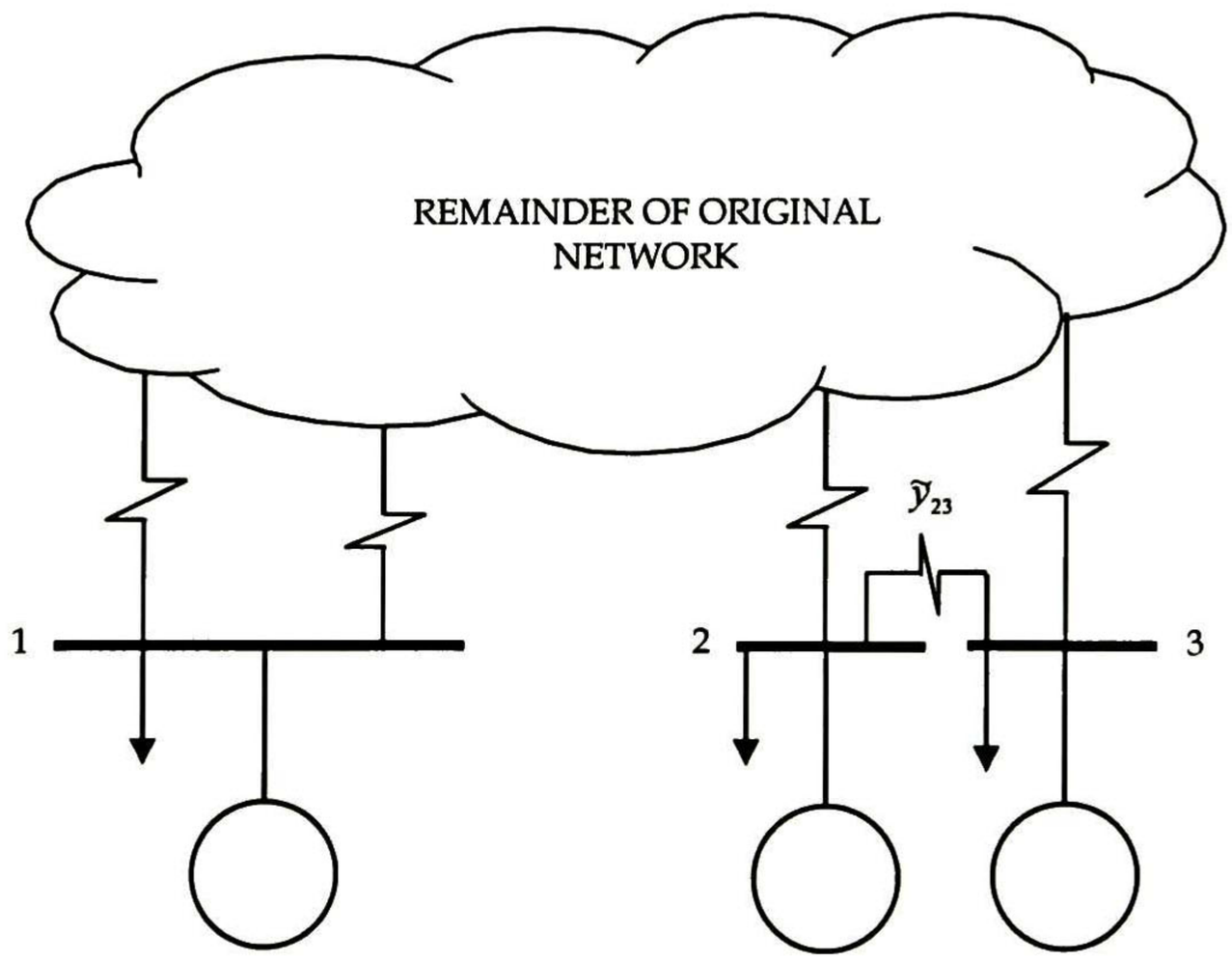



Fig. 2.4a Formation of coherent generator buses in original network.

Step 2.

Generally, the generator terminal buses are connected radially by a step-up transformer to the remains of the power system. In spite of this, many times the low voltage bus may possibly have been removed by mixing the transformer reactance with the generator internal reactance. After this condition, some non-radial buses could be included inside the coherent group and a common branch could tie them. (e.g., the branch between buses 2 and 3 in the Fig. 2.4b). The purpose in this second stage is to identify this condition and to eliminate the intragroup branch by substituting it by an equivalent shunt admittance. Let's consider the current flow in the branch between buses 2 and 3:

$$\bar{I}_{23} = (\bar{V}_2 - \bar{V}_3)\bar{Y}_{23} \tag{2.17}$$

As $\bar{V}_2 / \bar{V}_3$ is constant, the current could be expressed as a linear function of both $\bar{V}_2$ or $\bar{V}_3$. Therefore, the effect of the branch can be substituted by a shunt admittance as it is shown in Fig. 2.4c.

Step 3.

Fig. 2.4d gives you an idea of how are aggregated the equivalent bus, the load and shunt admittances of coherent buses. It is important to keep in mind, that the generation and load do not suffer any changes due to the transfer. As well, if a non-linear load representation is applied in such case the constant MVA, constant current and constant impedance load components will be transferred separately and maintained disconnected.

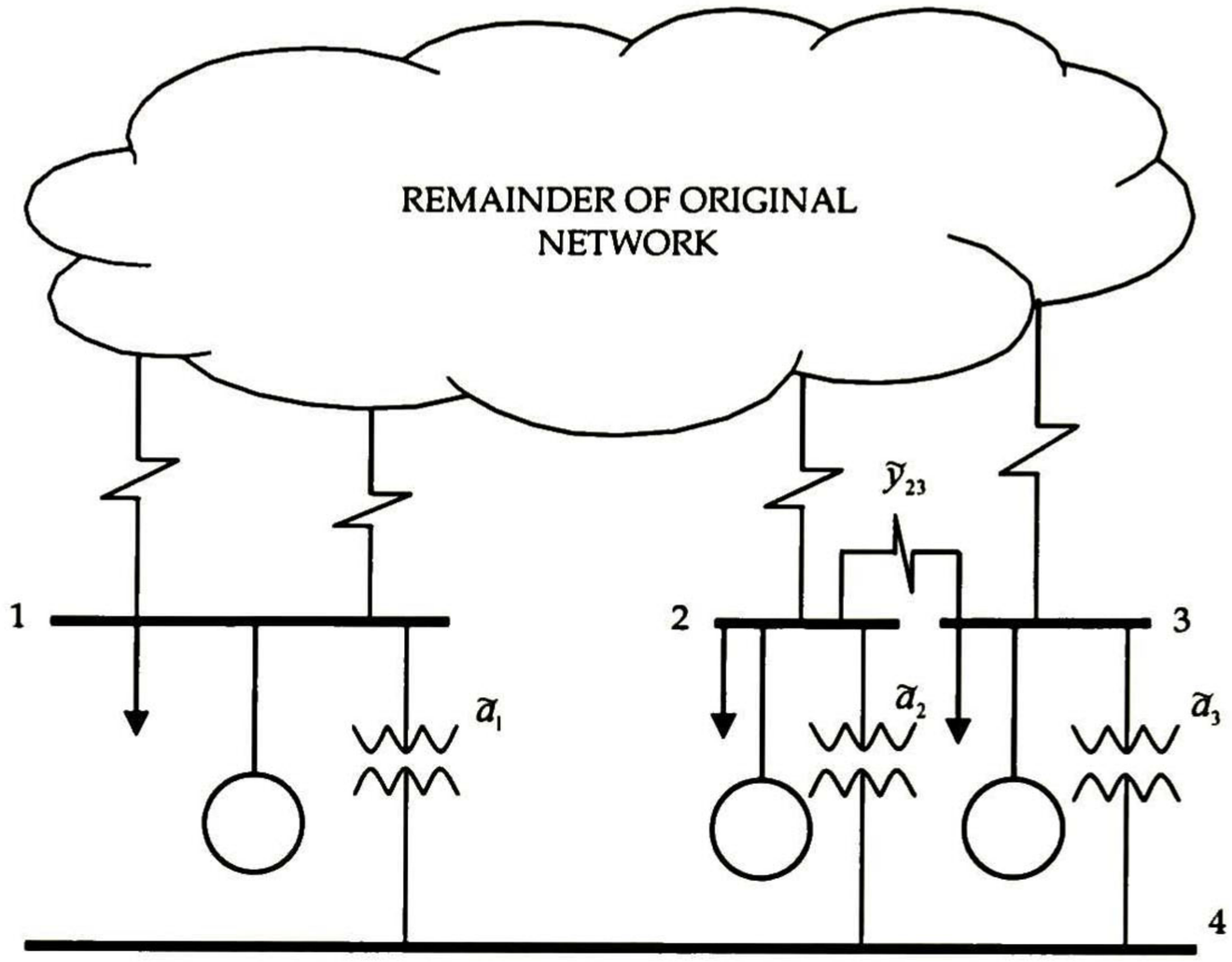



Fig. 2.4b Coherent generator buses are connected to an equivalent bus through ideal transformers with a complex ratio.

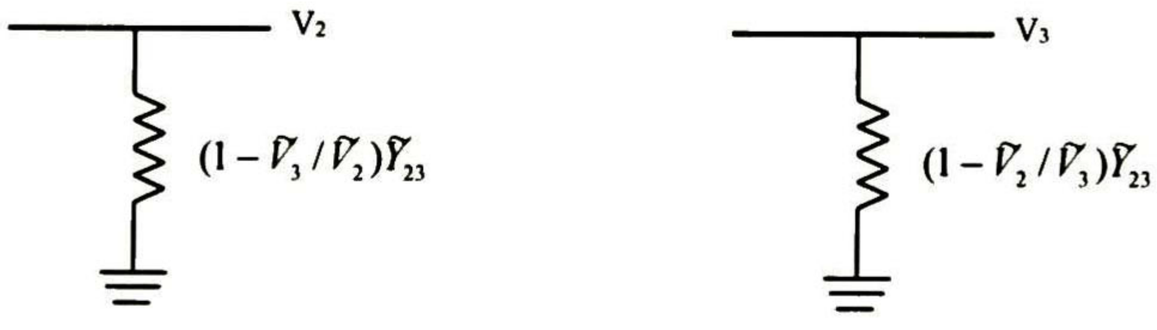



Fig. 2.4c Shunt admittances

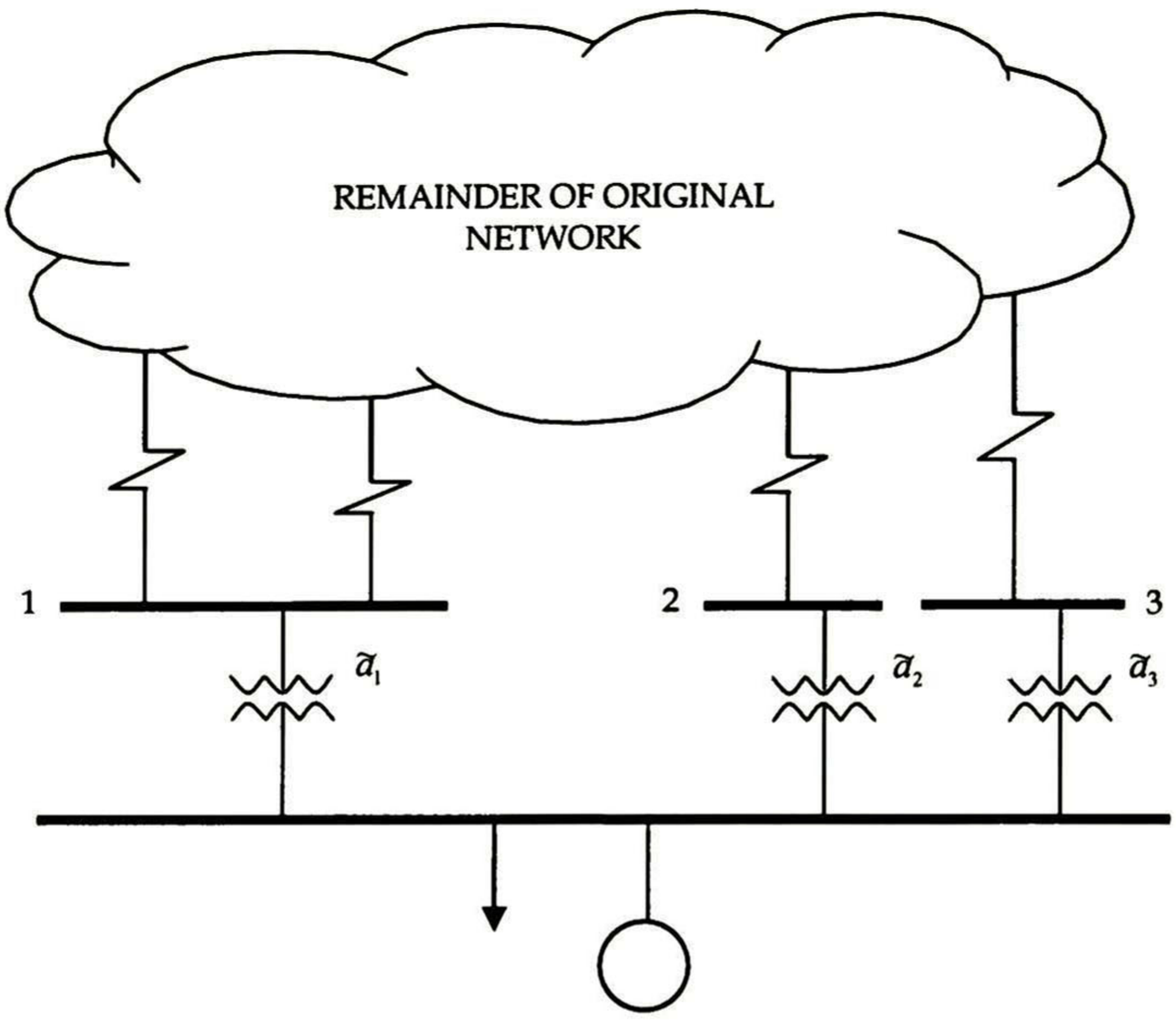



Fig. 2.4d Generation, load and shunt admittances on original buses are transferred to the equivalent bus.

Step 4.

The original coherent buses are removed by a sequences of fusion of the original branch and the ideal transformer (Fig. 2.4e). Let remark that when several original branches are linked to the eliminated bus, like in bus 1, the ideal transformer must be combined with each of them.

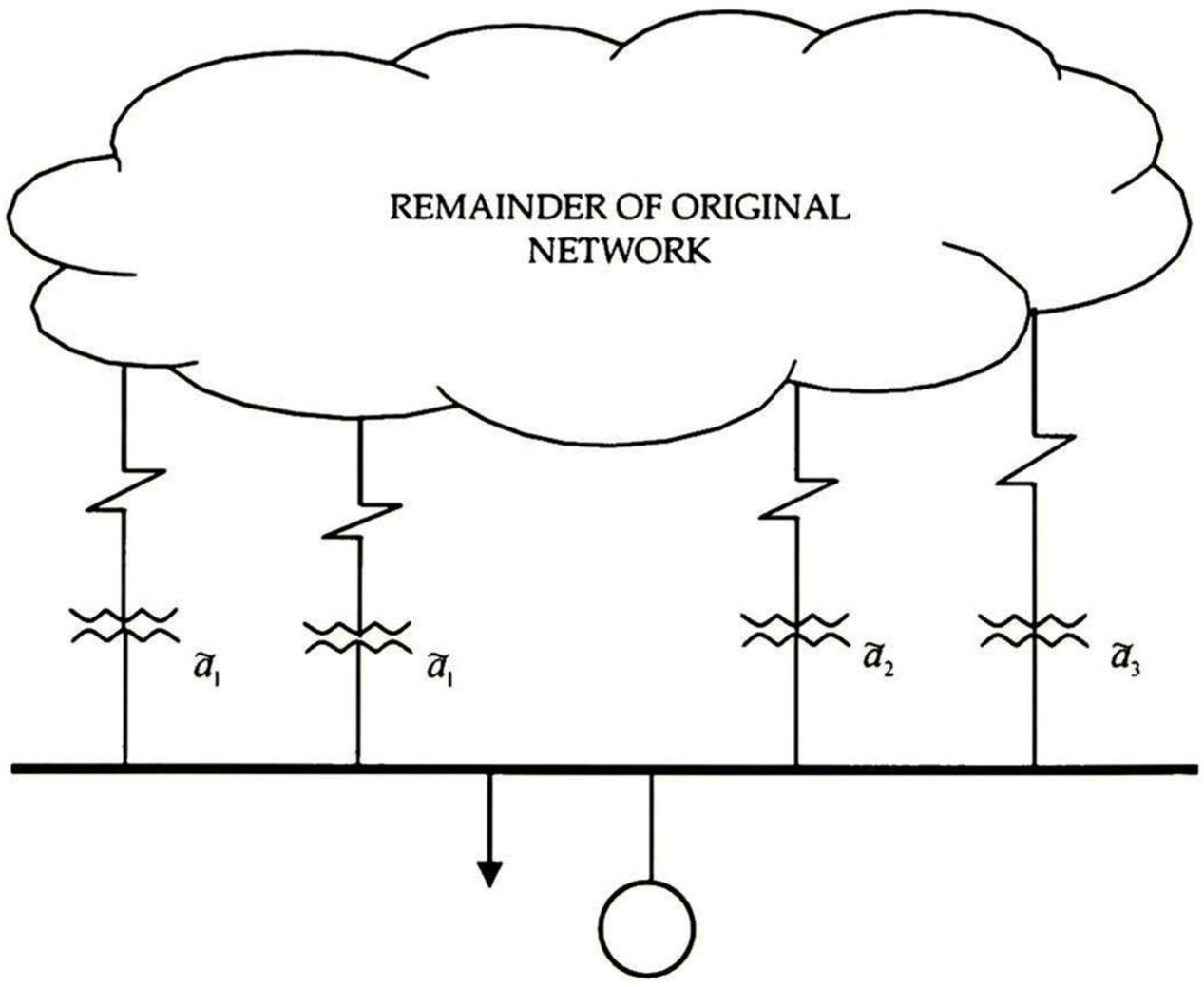



Fig. 2.4e Original generator terminal buses removed by sequence fusion of ideal transformers with original branches.

The next step is not mandatory; it could be done optionally in order to acquire additional simplification, with the exception that some accuracy will be lost.

Step 5.

The phase shifts in ideal transformers are replaced by compensating shunt admittances. These shunt admittances are calculated so that the power flow from buses at the ends of the branch are conserved. This step has been helpful for introducing the equivalent inside transient stability programs, which do not model phase shifters. The phase shifts, which are introduced into the equivalent lines, are directly associated to the angle of the voltage $\bar{V}_i$ ; hence depending on the selected value of $\bar{V}_i$ it will affect the accuracy of any power system.

## 2.7 Dynamic Aggregation of Generating Unit Models.

The technique of forming a dynamic equivalent for a given group of generating units presumes just that these units are coherent and connected to a common bus. Coherent generating units have the same speed $\omega$ and the same terminal voltage $\bar{V}_i$ for the reason that they are connected to a common bus because of network reduction [15, 17].

The efficient relations linking the mechanical and electrical output of an individual generating unit and its speed $\omega$ and terminal voltage $\bar{V}_i$ , are represented by the block diagram in Fig. 2.5, where these are considered as input variables.

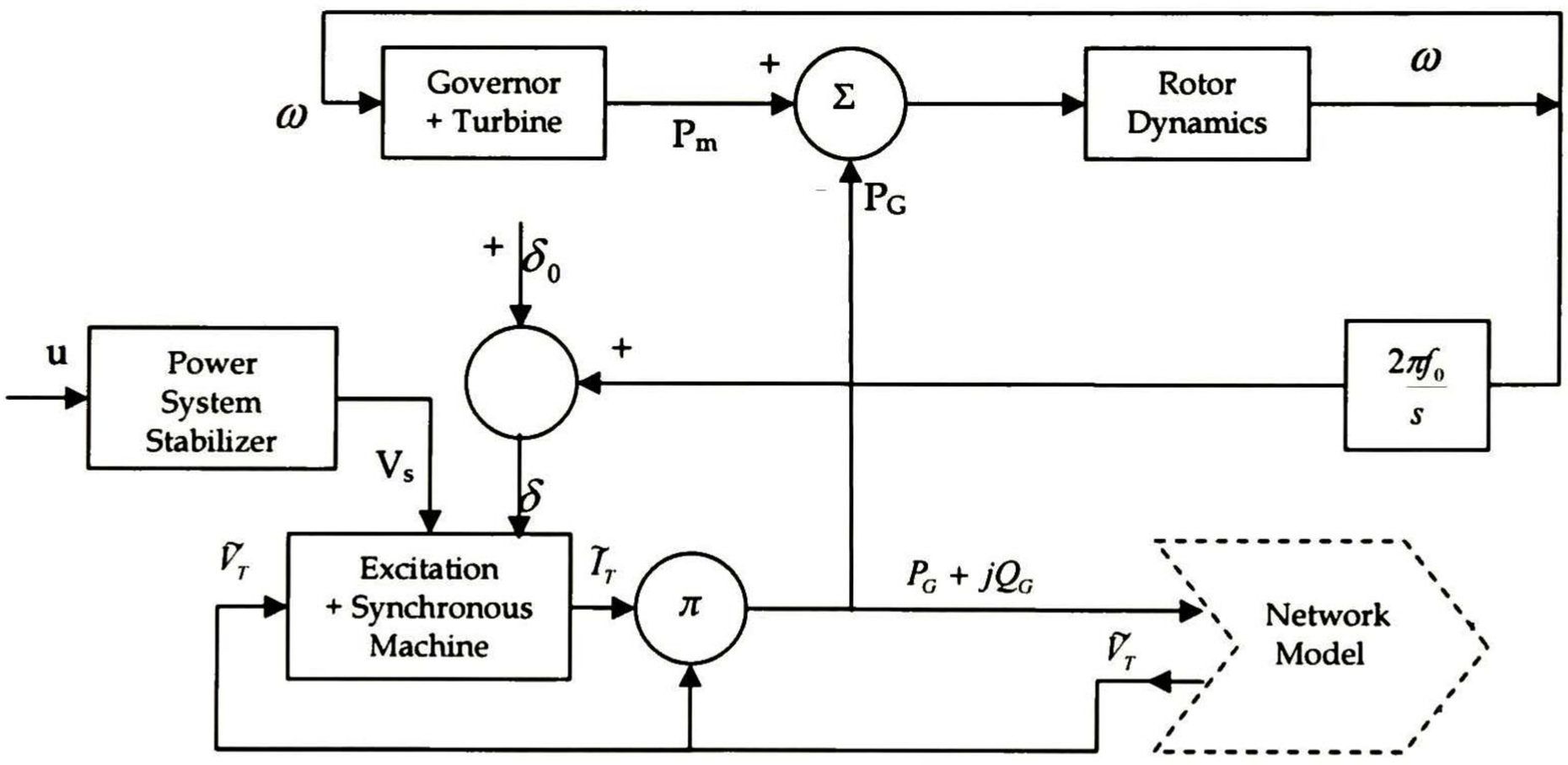



Fig. 2.5 Generating unit model.

Definition of variables:

| | |
|---|---|
| $\omega$ | frequency deviation. |
| $P_m$ | total mechanical power in p.u. |
| $P_G$ | total active power output in p.u. |
| $Q_G$ | total reactive power output in p.u. |
| $\bar{V}_T$ | terminal voltage. |
| $\bar{I}_T$ | terminal current. |
| u | power system stabilizer input signal. |

$V_s$ power system stabilizer output.

$\delta$ angle of machine internal voltage.

The main purpose of the process is to specify the features of this equivalent model, given the model of each individual unit. This is done by considering separately the rotor dynamics, the turbine-governor model, the excitation system model, the synchronous machine model, and the power system stabilizer model. The linear parameters of each equivalent model are numerically settled to get a minimal error between its transfer function and the sum of the transfer functions of the individual units. The error to be minimized is the total relative variation of the square magnitudes, for particular discrete frequencies. The transfer functions to be approximated are indicated in the Table 2.1.

Table 2.1 Open-loop transfer functions to be approximated by the equivalent models

| Open-Loop Transfer Function | Equivalent Model |
|---|---|
| $\omega / (\Sigma P_M - \Sigma P_G)$ | Rotor Dynamics |
| $\Sigma P_M / \Delta\omega$ | Governor + Turbine |
| $\Sigma T_T / V_T$ | Excitation System + Synchronous Machine |
| $V_S / u$ | Power System Stabilizer |

## 2.7.1 Aggregation Method.

### 2.7.1.1 Rotor Dynamics Aggregation.

First, the rotor dynamics will be considered, so the mechanical equation for one machine is given as

$$2H_j \frac{d\omega_j}{dt} = P_{Mj} - P_{Gj} - D_j\omega_j \tag{2.18}$$

where

$\omega$ is the speed deviation from the synchronous machine.

$H$ is the inertia constant (generator + turbine).

$P_M$ is the mechanical power.

$P_G$ is the electromechanical power.

$D$ is the damping constant.

$j$ is the machine subscript.

Let us notice that all parameters are being referred to the same system base MVA. Under coherency assumption, all machines of the group have the same speed deviation and for the equivalent is possible to have

$$\left(\Sigma_j 2H_j\right)\frac{d\omega_j}{dt} = \Sigma_j P_{Mj} - \Sigma_j P_{Gj} - \left(\Sigma_j D_j\right)\omega_j \tag{2.19}$$

Check that the equivalent inertia constant is the sum of the individual inertia constants and that the equivalent-damping factor is the sum of the individual damping factors.

### 2.7.1.2 Aggregation of the Synchronous Machine.

When both classical and detailed models are in a coherent group, the aggregation will be done separately, so the classical models (not including the excitation control) are aggregated to form an equivalent classical unit and the same process will be repeated for the detailed models creating an equivalent detailed generating unit.

Assuming coherency, two considerations will be made:

- ➔ The difference of rotor angles between machines of a coherent group continues constant.
- ➔ The terminal voltages are the same for each machine of the group, as they are linked in parallel in the same bus after reducing the coherent buses.

Using these two assumptions, it is feasible to represent a dynamic equivalent as a two-axes model with one field winding in the direct axis and one damping winding in the quadrature axis. This model is represented by Fig. 2.6.

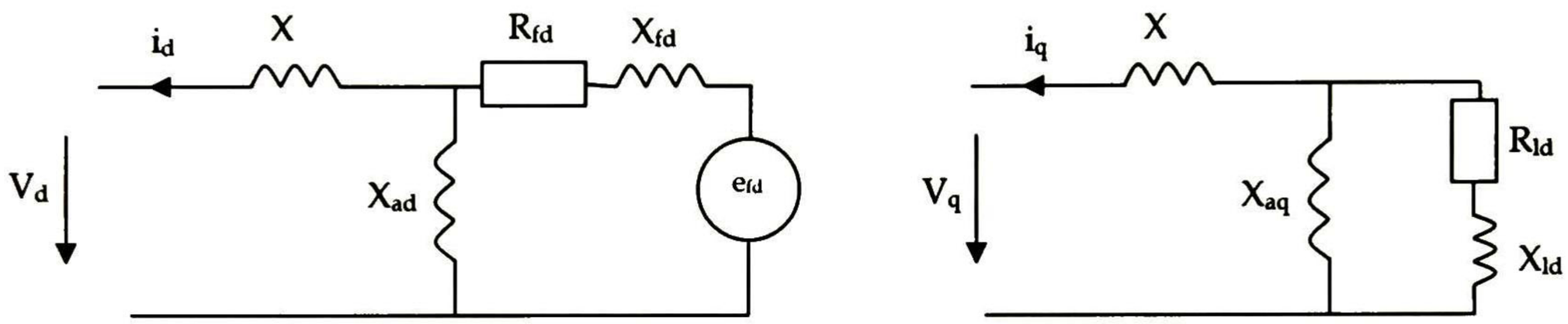



Fig. 2.6 Two-axes model of the synchronous machine.

where: $i_d$ is the direct axis of stator current component.
$i_q$ is the quadrature axis of stator current component.
$V_d$ is the direct axis terminal voltage.
$V_q$ is the quadrature axis terminal voltage.

Note: All values are per unit.

So then, the total electromagnetic power output in p.u. $P_G$ of the coherent group is given by the next equation:

$$P_G = \Sigma_j (V_{qj} i_{qj} + V_{dj} i_{dj}) \tag{2.20}$$

where the terminal voltage $V$ and the stator current $i$ for each machine are expressed in its own reference axes, which are represented by $d$ and $q$.

Due to the terminal voltage is common, the total electric power could be denoted by:

$$P_G = V_Q \Sigma i_{Qj} + V_D \Sigma i_{Dj} \tag{2.21}$$

where the subscripts $D$ and $Q$ correspond to the components on an arbitrary pair of orthogonal axes. In addition, the equivalent machine can be represented by a two-axes model, thus its electric power output is:

$$\overset{*}{P}_G = V_Q \overset{*}{i}_Q + V_D \overset{*}{i}_D \tag{2.22}$$

with

$$\begin{bmatrix} \overset{*}{i}_D(s) \\ \overset{*}{i}_Q(s) \end{bmatrix} = \begin{bmatrix} 0 & \overset{*}{Y}_{DQ}(s) \\ \overset{*}{Y}_{QD}(s) & 0 \end{bmatrix} \begin{bmatrix} V_D \\ V_Q \end{bmatrix} + \begin{bmatrix} \overset{*}{Y}_{DF} \\ 0 \end{bmatrix} e_{FD} \tag{2.23}$$

(the symbol * indicates equivalent variables and parameters).

The aggregation of the synchronous machine will be done in two simple steps as follows:

- ✓ The first step only consists in calculate the position of the axes.
- ✓ On the second step the parameters of the equivalent model are adjusted independently for each axis fitting the operational admittance.

2.7.1.3 Aggregation of the Excitation System Model.

For the excitation system model, the aggregated transfer function that will be approximated is a biased sum of the transfer functions for the individual excitation systems. The weighting factor for an individual excitation system depends on the parameters of the synchronous machine to which it is connected, and on the parameters of the equivalent synchronous machine. The weighting factors consider the fact that the field voltage of larger units has more influence on terminal voltage from the coherent group than the field voltage from small units.

2.7.1.4 Aggregation of the Power System Stabilizer Model.

A power system stabilizer introduces an adjustment to the reference voltage of the excitation system. It is noticed that coherent groups with generator that have different transfer functions, as steam or hydro generating unit, or power systems stabilizer with not the same input signal, could not be aggregated into a simple equivalent unit.

In view of the fact that the input signal $u$(s) for the entire power systems stabilizer must be equal in a coherent group, the relation for the equivalent is:

$$\frac{\Delta e_{DF}(s)}{u(s)} = \overset{*}{G}_E(s) \cdot \overset{*}{G}_s(s) \tag{2.24}$$

where

$\overset{*}{G}_E(s)$ is the linear transfer function of the equivalent excitation system.

$\overset{*}{G}_S(s)$ is the linear transfer function of the equivalent power system stabilizer.

## 2.8 Load Buses Reduction.

At this stage, it is merely considered the reduction of buses which have constant impedance loads. Some years ago, the most common techniques that had been applied considerably for reduction of constant impedance loads for the purpose of solving load flow and transient stability were the *Ward-Hale* or *Gaussian elimination method* and the *REI* (*Radial, Equivalent and Independent network*) method [6]. Owing to the network that represents the original power system is certainly very sparse, the Ward-Hale elimination technique reduces the number of buses, but there is not certainty that the number of lines will be reduced too, and this is really significant since the overall computing time.

Nowadays, sparsity techniques have been applied successfully to the network reduction problem in order to minimize the number of branches which are introduced into the equivalent network [6, 15]. In this sparsity oriented reduction, it is necessary to identify the key buses which have propensity to be buses which either have an important number of connections or buses that connect sub-areas that have few connections to the rest of the system. The procedure which has been the most effective for identifying key buses is based in the bus elimination order using a sparsity oriented scheme and finishing the bus elimination when the number of terms in the equivalent admittance matrix begins growing instead of decreasing.

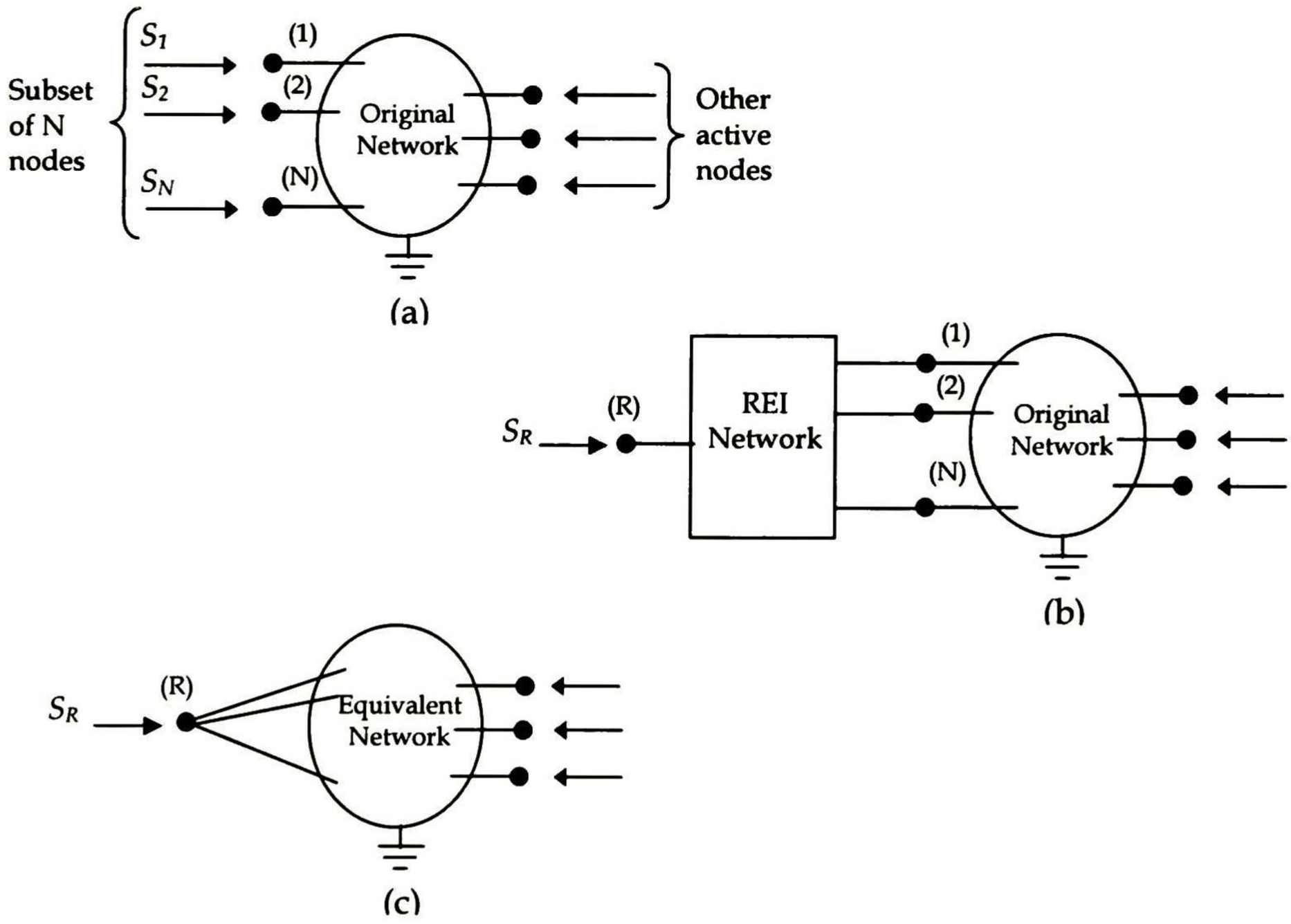



Fig. 2.7

(a) Original network showing group of nodes that will be converted into a REI equivalent.
(b) REI network connected to original network.
(c) Equivalent network after elimination of passive nodes.

To explain the basic idea of the REI technique it is helpful to add some diagrams (Fig. 2.7). Fig. 2.7a shows a network at an operating point, which has been established previously, and a subset of $N$

active nodes of the network with injections $S_1$, $S_2$,..., $S_N$ (complex power injections) will be converted into a REI equivalent. The first step is shown in Fig. 2.7b where a REI network is connected to the $N$ nodes. After the connection, the $N$ nodes will be passive nodes. The REI network has one active node $R$, with injection $S_R$, as well as the $N$ connecting nodes.
The REI network has no specific internal structure but it is formed of passive linear elements and has no connection to ground [6].

The injection $S_R$ is equivalent to the algebraic sum of the $N$ known injections $S_i$. On the REI network, the real and reactive power losses must be zero and its connection must not change the electrical conditions of the original network at the solution point. Hence, voltages $V_i$ from the original nodes and the flows from the REI network into the connecting nodes must be the same as before. In view of the fact that the internal nodes plus the $N$ connecting nodes are passive, they can all be left out without affecting the condition at remaining nodes of the original network. So, when the elimination of the $N$ connecting nodes from the REI network has been done, the shape of the modified network (Fig. 2.7c) will be exactly the same as the original network at the operating point. $S_R$ will substitute the $N$ power injections, thus the correlation between the powers input-output of the equivalent network will be also the same as for the original network. It is always possible to create a REI network that satisfy the previous conditions. The REI equivalent design must be in the way so that the original network does not suffer any changes and the network design must always obey the Kirchhoff's Laws.

**SYNOPSIS.**

This chapter summarizes the key concepts to understand the stability problem in power systems, such as power oscillations, equal-area criterion, load modelling and synchronous machine models are presented. Additionally, are described the main reasons to develop a dynamic equivalent for any electric grid, the most important methods to derive power system dynamic equivalents for transient stablity studies and other main impressions about dynamic aggregation and load buses reduction.

**REFERENCES.**

1. Pavella M. and Murthy, *Transient Stability of Power Systems*, John Wiley & Sons, 1995.
2. Graham Rogers, *Demystifying Power System Oscillations*, IEEE Computer Application in Power, Vol. 9, No. 3, pp 30-35, July, 1996.
3. Soos Antal, *An Optimal Adaptive Power System Stabilizer*, Ph.D Thesis, University of Calgary, Department of Electrical and Computer Engineering, Calgary, Alberta, October, 1997.
4. Nasar S.A. and Trutt, F.C., *Electric Power Systems*, CRC Press, 1999.
5. Lee, Stephen and Schweppe, Fred, *Distance Measures And Coherency Recognition For Transient Stability Equivalents*, IEEE Power Engineering Society, paper 73 TP 092-4, pp 1550-11557, January, 1973.

6. Tinney, W. F. and Powell, W. L., *The REI Approach To Power Network Equivalents*, Power Industry Computer Applications Conference, Bonneville Power Administration (BPA), pp 314-320, 1977.
7. Djukanovic Miodrag, Sobajic Dejan J. and Pao Y. H., *Artificial Neural Network Based Identification Of Dynamic Equivalents*, Elsevier Science Ltd, Electric Power Systems Research 24, pp 39-48, 1992.
8. Yao-nan Yu and M.A. El-Sharkawi, *Estimation Of External Dynamic Equivalents Of A Thirteen-Machine System*, IEEE Transaction on PAS, Vol. PAS-100, No. 3,pp 1324-1332, March 1981.
9. Pavella M. and Belhomme R., *A Composite Electromechanical Distance Approach To Transient Stability*, IEEE Transaction on Power Systems, Vol. 6, No. 2, pp 622-631, May, 1991.
10. Wilfert H., Voigtländer K. and Erlich I., *Dynamic Coherency Identification Of Generators Using Self-Organising Features Maps*, Elsevier Science Ltd, Control Engineering Practice 9, pp 769-775, 2001.
11. Lawler J. S. and Schlueter R. A., *Computational Algorithms For Constructing Modal-Coherent Dynamic Equivalents*, IEEE Transactions on PAS, Vol. PAS-101, No. 5, pp 1070-1080, May 1982.
12. Price W., Ewart D., Gulachensky E. and Silva R., *Dynamic Equivalents From On-Line Measurements*, IEEE Transactions on Power Apparatus and System, Vol. PAS-94, No. 4, pp 1349-1357, July/August, 1975.
13. Kundur P., *Power System Stability and Control*, McGraw-Hill, Inc., 1994.
14. IEEE Task Force Report, *Load Representation For Dynamic Performance Analysis*, Paper 92WM126-3 PWRD, presented at the IEEE PES Winter Meeting, New York, January, 1992.
15. Podmore Robin and Germond Alain, *Development Of Dynamic Equivalents For Transient Stability Studies*, Electric Power Research Institute (EPRI), Research Project 763, EPRI EL-456, April, 1977.
16. Rusche P, Schlueter R. A., et. al, *Modal-Coherent Equivalents Derived From An RMS Coherency Measure*, IEEE Transactions on Power Apparatus and System, Vol. PAS-99, No. 4, July/August, 1980.
17. *Coherency Based Dynamic Equivalents For Transient Stabuility Studies*, Electric Power Research Institute (EPRI), Final Research Project RP904, December, 1974.
18. Di Caprio U., *Conditions For Theorical Coherency In Multimachine Power Systems*, $7_{th}$ IFAC World Congress, June, 1978.

# Chapter 3

# ARTIFICIAL NEURAL NETWORKS

*Man cannot make the principles, he can only discover them*
*T. Paine*

## 3.1 Introduction.

The human brain is a network consisting of approximately 2.5 billion processors, named *neurons*, which are the essential units of the nervous system. From a classical standpoint, a neuron is a simple processing unit that receives and combines signals from many other neurons via filamentary input paths [1] called *dendrites*; if dendrites get together they will take the shape of a *dendritic tree*, and these dendritic trees are linked to the *soma*, which is the main body of the nerve cell. The external periphery of the cell is the *membrane*. There is another branchlike structure called *axon* and other structures called *synapses*, which connect axons and dendrites from a neuron to those of another one. These synapses can excite or inhibit travelling signals between neurons. When synapses are excited above a certain level, the threshold level, the neuron fires producing an output signal. This signal is sent to other neurons through the synapses, and these neurons produce their own firing actions, Fig. 3.1.

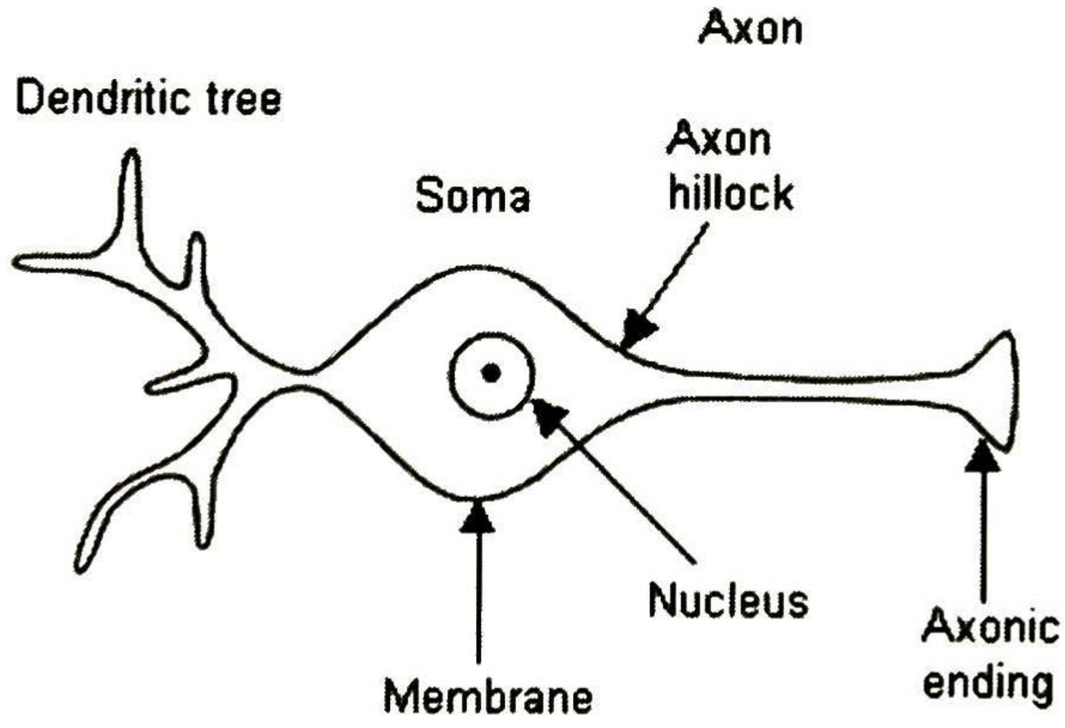



Fig. 3.1 Representation of a neuron.

*Artificial neural networks* (ANNs) mimic the brain and they are modelled as its physical architecture. ANNs consist of many interconnected neurons, or processing elements, with familiar characteristics, such as *input, synaptic strength, activation function, output* and *bias* [1]. The processing of an artificial neuron is characteristically constrained to a non-linear function, which is able to emulate the firing action of a real neuron. Making use of the previous conceptions, a mathematical ANN model is defined as a direct graph including the following properties [2]:

i. A state variable $x_i$ is associated with each node $i$.
ii. A real-valued weight $w_{ik}$ is associated with each link $(ik)$ between two nodes $i$ and $k$.
iii. A real-valued bias $v_i$ is associated with each node $i$.

iv. For each node *i*, a transfer function $f_i[x_k, w_{ik}, v_i, (k \neq i)]$ is defined, which determines the state of the node as a function of its bias, the weight of its incoming links, and the states of the nodes connected to it by these links.

Neurons and synapses are called *nodes* and *links*, respectively, and the bias is known as the *activation threshold*. Regularly, the transfer function could be written as $f\left(\sum_k w_{ik} x_k - v_i\right)$, where $f(\bullet)$ is a discontinuous step function. The fundamental features of neural networks may be divided into two groups: the *architecture* and *neurodynamics* or functional properties. The architecture is the number of artificial neurons in the network and their interconnectivity; it will define the network structure. The neurodynamics of neural networks are how the neural network learns, recalls, associates, and continuously compares new information with existing knowledge, how it classifies new information and how it develops new classification if required. Collective and synergistic computation, robustness, learning and asynchronous operation are some of the characteristics of ANNs [1].

### 3.2 Artificial Neural Network Classification.

There are two categories in which ANNs could be classified, and they are according to their structure and *learning algorithms*. In terms of their structure, ANNs can be classified into two categories: *feedforward networks* and *recurrent networks*. There are two types of learning algorithms: *supervised learning algorithm* and *unsupervised learning algorithm*, also known as *self-organizing*. This learning algorithm concerns the detection of unlabeled patterns of a given training set, there are no outputs known a priori, and the basic scheme is to optimise some criterion. The purpose is to find out or to classify features or irregularities in the training data without using external aid. Many problems that require an algorithm to cluster, approximate and compress given information take advantage of these unsupervised algorithms. The most common unsupervised learning algorithms are

- Winner-takes-all learning algorithm.
- Adaptive resonance theory (ART).
- Hebbian learning.
- Self-organising feature maps (SOFM).
- Kohonen's SOFM.

An application of one of these unsupervised algorithms (Winner-takes-all), especially for data compressing, is a learning vector quantizer (LVQ).

Conversely, supervised learning assumes that for every input the output is known a priori. There are variants of the supervised algorithm, such as:

- Competitive learning.
- Cooperative learning.
- Reinforced learning.
- Error-correcting learning.
- Markovian (stochastic) learning.

The general idea about how the ANNs learn during supervised learning algorithm is in this way. When an input is applied, the obtained output is compared with the desired or target output. From this comparison, if they do not match, an error signal is generated and used to make parametric adjustments of the neural network until the desired output becomes roughly equal to the target output. These adjustments are made based on an optimisation algorithm.
There is a vast and mature field of numeric optimisation techniques; only the types of search methods that are important for artificial neural networks are described in the following.

### 3.2.1 Newton's Method.

The method consists of an iterative technique for solving an equation of the form $P(x)=\theta$, however it can be extended for the optimisation (minimization) of variable functions, as it is next described [3]. Let us consider the quadratic approximation of the function $f(X)$ at $\mathrm{X}=\mathrm{X_i}$ using the Taylor's series expansion

$$f(X) = f(X_i) + \nabla f_i^T (X - X_i) + \tfrac{1}{2}(X - X_i)^T * [J_i](X - X_i) \tag{3.1}$$

where $[J_i] = [J]|x_i$ is the matrix of second partial derivatives (Hessian matrix) of $f$ evaluated at a point $X_i$. By setting the partial derivatives of equation (3.1) to zero,

$$\frac{\delta f(X)}{\delta x_j} = 0 \qquad\qquad j = 1,2,\ldots,n \tag{3.2}$$

Equations (3.2) and (3.1) give rise to

$$\nabla f = \nabla f_i + [J_i](X - X_i) = 0 \tag{3.3}$$

if $[J_i]$ is non-singular, Equations (3.3) can be solved to obtain an improved approximation $(X = X_{i+1})$ as

$$X_{i+1} = X_i - [J_i]^{-1} \nabla f_i \tag{3.4}$$

Since higher-order terms have been neglected in Eq. (3.1), Eq. (3.4) is to be used iteratively to find the optimum solution $X^*$
The sequence points $X_1, X_2, \ldots X_{i+1}$ can be shown to converge with the actual solution $X^*$ from any initial point $X_1$ sufficiently close to the solution $X^*$, assuming that $[J_1]$ is non-singular. It can be seen that Newton's method uses the second partial derivatives of the objective function (in the form of the matrix $[J_i]$), hence it is a second-order method.

### 3.2.2 Gradient Method.

Contrary to the preceding method, this approach uses only the first derivatives of the objective function in calculations. *Gradient-based* algorithms are the most common and important non-linear local optimisation techniques [3]. The gradient is the vector at a point $x$ that gives the (local) direction of the greatest increase in $f(x)$ and is orthogonal to the contour of $f(x)$ at $x$. For maximization, the

search direction is just the gradient (named "*steepest ascent*"); for minimization, the search direction is the negative of the gradient ("*steepest descent*")

$$s^k = -\nabla f(x^k) \tag{3.5}$$

In the steepest descent at the *k*-th iteration, transition from point $x^k$ to another point $x^{k+1}$ can be viewed as given by the following expression:

$$x^{k+1} = x^k + \Delta x^k = x^k + \lambda^k s^k = x^k - \lambda^k \nabla f(x^k) \tag{3.6}$$

where

$\Delta x^k$ is the correction vector from $x^k$ to $x^{k+1}$

$s^k$ is the search direction; the direction of steepest descent

$\lambda^k$ is the scalar that determines the step length in direction $s^k$

The negative of the gradient gives the direction for minimization but not the magnitude of the step to be taken. It is assumed that the value of $f(x)$ is continuously reduced. Equation 3.6 must be applied repetitively until the minimum is reached. At the minimum, the value of the vector gradient will be zero.

### 3.2.3 Levenberg-Marquardt Method.

This approach was proposed, independently, by Levenberg (1944) and Marquardt (1963), and is designed specially for non-linear least squares [3, 13]. This method guarantees that the Hessian matrix is positively defined and well-conditioned. This is done by modifying the Hessian matrix $H(x)$ of $f(x)$ on each step of the search. The process adds elements to the diagonal elements of $H(x)$,

$$\tilde{H}(x) = [H(x) + \beta I] \tag{3.7}$$

where $\beta$ is a positive constant greatly adequate to have $\tilde{H}(x)$ positive definite when $H(x)$ is not. To ascertain the $\beta$ value to be used it is necessary to estimate the smallest eigenvalue (most negative) of $H(x)$ and make $\beta > -\min\{\alpha_1\}$, where $\alpha_1$ is an eigenvalue of $H(x)$. Note that if $\beta$ is great enough, $\beta I$ can overpower $H(x)$ and the minimization comes close to a steepest-descent search.

### 3.2.4 Simulated Annealing - Based Global Search.

Simulated Annealing (SA) is a Monte Carlo - stochastic - method for global optimisation where the space is searched in a random rule to avoid finishing in local minima and has demonstrated to be exceptionally useful in locating the global minimum of objective or cost functions derived from complex non-linear systems. SA was first proposed by Kirkpatrick, Gelatt and Vecchi [11] in 1983 as a combinatorial optimisation algorithm. The abstract notion of the process is ruled by the theory of Markov chains, and the simulation is possible via the Metropolis algorithm [12]. The central theory of this method is how a solid-state material is heated up to a temperature waiting for reaching an

amorphous liquid shape. After that, it is cooled gradually and according to a specific schedule, the temperature decreases. If the initial temperature is high enough to ensure a sufficient random state, and if the cooling is slow enough to ensure that thermal equilibrium is reached at each temperature, then the atoms will arrange themselves in a pattern that closely resembles minimum global energy. This combinatorial optimisation algorithm has a primary feature, a generation mechanism which selects a solution *j* from the neighborhood $S_i$ of the solution *i*. The method does not require gradient information, so it is appropriate for a wider variety of functions than the stochastic methods.

### 3.2.5 Genetic Algorithms.

Genetic algorithms (GA) are global (stochastic) optimisation algorithms based on the mechanics of natural selection and natural genetics and were initially formulated by Holland (1975) [14, 16]. GA originally operated on a binary level and are extremely similar to nature, where the information is coded in four different bases ("A", "G", "C", "T") on the DNA. Genetic algorithms start with an initial set of random solutions called *population*. Each individual in the population is named a *chromosome*, representing a solution to the problem. A chromosome is a set of *genotypes*, which store the characteristics of solutions. The chromosomes grow due to successive iterations labelled as *generations*. The objective function (fitness measuring criterion) determines the suitability of each solution. Founded on these values, some of them are selected for reproduction. Genetic operators are applied on these (selected) parent chromosomes and new chromosomes (*offspring*) are generated. The operators frequently employed in GA are selection/reproduction, crossover, and mutation which are used to generate a new population. Some of the parents form offspring by rejecting others to keep the population size constant. Fitter chromosomes have higher probabilities of being selected. After several generations, the algorithm converge to the best chromosome, which hopefully represents the optimum or sub optimum solution to the problem.

## 3.3 Feedforward Networks.

In this kind of ANNs neurons are usually clustered into layers. Signals run from the input layer through the output layer due to unidirectional connections, the neurons being connected from one layer to the next, but not to the same layer. The *multi-layer feedforward network* is responsible for most of the successful applications of neural networks [5] and is certainly the most commonly used neural network. The best example of feedforward network is the *multi-layer perceptron* (MLP) network. However, there are many other types of feedforward networks as the *learning vector quantization* (LVQ) network, the *cerebellar model articulation control* (CMAC) network and the *group-method of data handling* (GMDH) network.

Other ANNs as *Perceptron* and *Adaline* which were developed before and were the heart for the multi-layer perceptron will be described in the following.

### 3.3.1 Perceptron Network.

This is the first arficial neural network model and was developed by Frank Rosenblatt [1,2,6,7,9,10] by the end of the 50's. This ANN is able to learn and recognize simple patterns; moreover, it is capable of separating many learning patterns into two classes. Its architecture is very simple. A perceptron is a linear gate, which consists of one exit neuron that can add up the entries, subtract the threshold and send the result through a transfer function which is a *unit step function*. Fig. 3.2 illustrates the single Perceptron model.

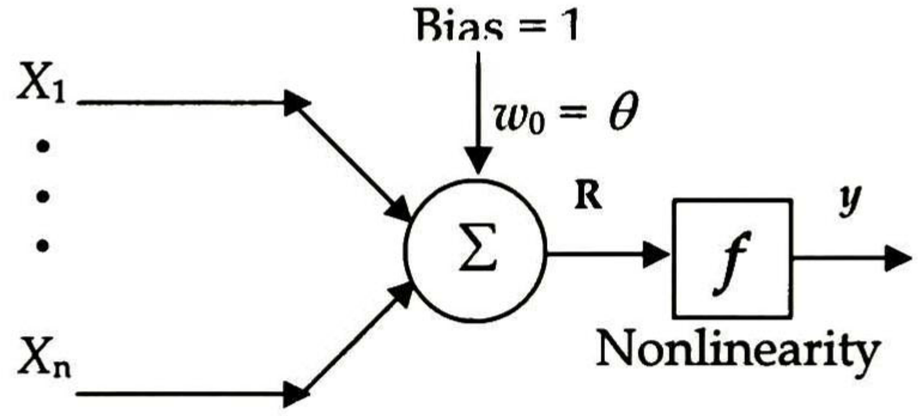



Fig. 3.2 Single Perceptron model.

The decision rule is +1 if the answer fits the pattern of one class or –1 if fits in other class. The output value must depend on the total entry, (which are the entries $x_i$ added up and multiplied by the weights $w_i$), and the threshold value $\theta$. It can be represented in a better way by the following equations.

$$y = \begin{cases} +1 \text{ if } \sum_{i=1}^{n} w_i x_i \geq \theta \\ -1 \text{ if } \sum_{i=1}^{n} w_i x_i < \theta \end{cases} \tag{3.8}$$

The technique to analyse the geometric reasoning for ANNs as perceptron is to represent on a map the "*decision regions*" created inside the multidimensional space in the network. In these regions patterns belonging to each class are visualized. The perceptron separates the regions using a hyper plane which equation is determined by the weights and the given activation threshold. Figure 3.3 can help to visualize the technique.

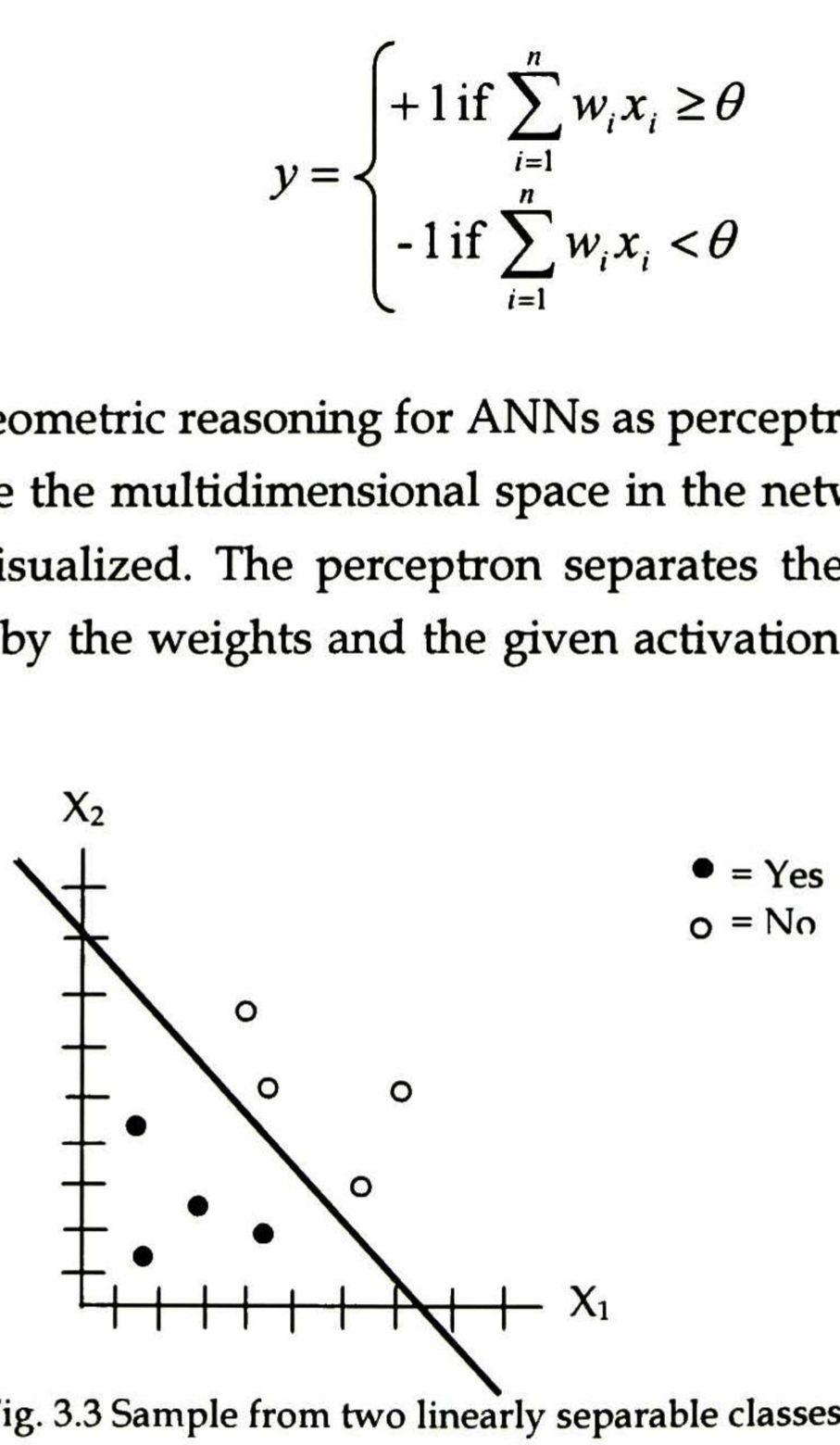



Fig. 3.3 Sample from two linearly separable classes.

For a bidimensional example, the perceptron network classifies data according to:

$$y = \begin{cases} \text{Yes} & \text{if } w_1 x_1 + w_2 x_2 \geq 0 \\ \text{No} & \text{otherwise} \end{cases}$$

Therefore, it separates the space into two halves, as it is shown in Fig. 3.3. The perceptron network possesses a learning algorithm which is one from the supervised type. During the training, the perceptron must adjust the weight and the threshold values, in which most of the times it is convenient to consider the threshold value as a weight $\theta = w_0$.

Perceptron Training Algorithm.

This learning procedure is described by five steps, which are summarized as follow.

- ✓ STEP 1. Initialise values.

  Start with randomly chosen weight values $w_i$, and the threshold value is taken as $\theta = -w_0$.

- ✓ STEP 2. Set the input and output values.

  Set the new input patterns $x_i = (x_1, \ldots, x_n)$ and the target output $d(t)$.

- ✓ STEP 3. Calculate the actual output value.

  The output value can be determined as follows:

$$y(t) = f\left[\sum_{i=1}^{n} w_i(t) * x_i(t) - \theta\right] \tag{3.9}$$

  where $f(x)$ is the unit transfer function.

- ✓ STEP 4. Update weights using the iterative relationship. This will be done using the next equation.

$$w_i(t+1) = w_i(t) + \alpha\left[d(t) - y(t)\right]x_i(t) \tag{3.10}$$

$$(0 \leq i \leq n-1)$$

  where,

  $d(t)$ corresponds to the target output.

  $y(t)$ represents the actual output.

  $\alpha$ corresponds to the gain factor or the learning rate.

The *learning rate* must be amid 0 and 1.0; it is adjusted to satisfy the quick learning requirement such as the stability and the estimation weights. This process is repeated until the errors produced by each pattern equal zero.

- ✓ STEP 5.
  Repeat step 2-4.

### 3.3.2 Adaptive Linear Neuron or Element (ADALINE) Network.

Almost simultaneously to the perceptron network, Adaline network was developed; Bernard Widrow was its designer. Its architecture is roughly the same as the perceptron network. The basic difference among them is concerning to the learning algorithm. Adaline network uses the *"Widrow-Hoff Delta rule"*, which is based on the error expression between the target output and the linear output obtained before applying the transfer function. Another special feature that makes the difference between Perceptron and Adaline is the transfer function. For the Adaline network the common transfer function is the *sigmoid function*. The main reasons motivating the use of an s-shaped sigmoid function are that it is continuous, monotonically increasing, invertible, everywhere differentiable and asymptotical; Fig. 3.4 depicts sigmoid function defined as

$$f(y) = \frac{1}{1+e^{-y}} \tag{3.11}$$

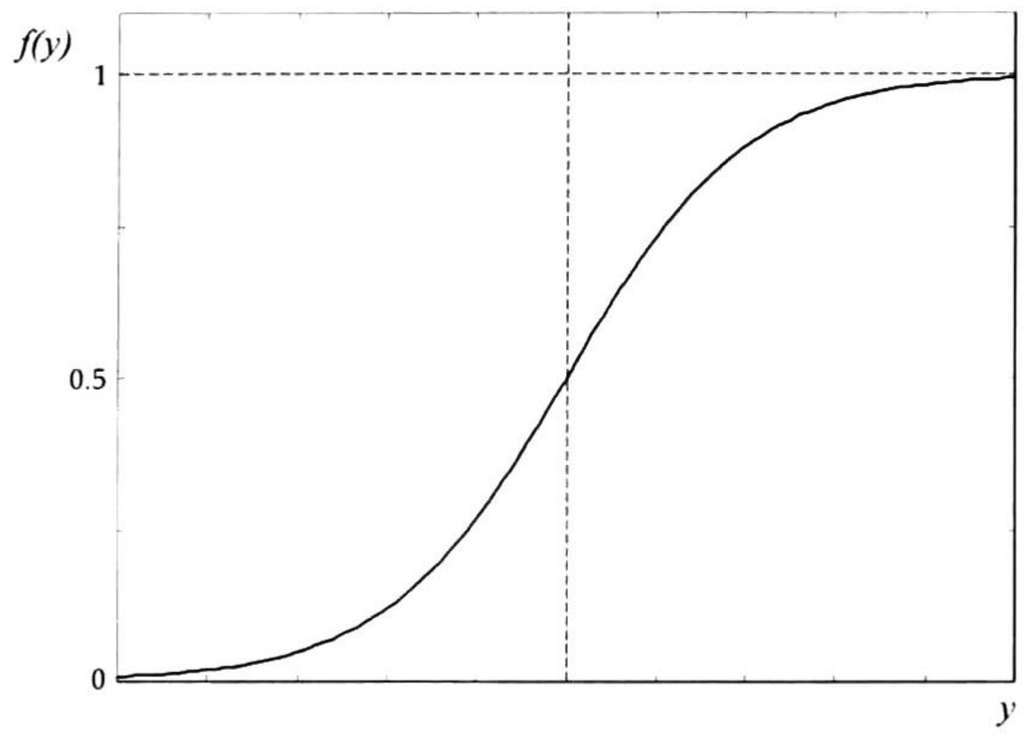



Fig. 3.4 A sigmoid function.

An Adaline is a simple system that accomplishes classification by adjusting weights in order to reduce the *mean squared error* (MSE) at every iteration [6]. This can be done using gradient descent. In other words, the Adaline model compares the actual output $R$ with the target output $T$; this is based on the mean-squared learning algorithm where the weights are adjusted and the error function is

$$E = T \quad R \tag{3.12}$$

The main objective is to adjust the weights so that the MSE is reduced through the next equation:

$$\frac{\delta w_{ij}}{\delta t} = \alpha \delta x_i \quad \frac{x_i}{x^2} \tag{3.13}$$

where,

$\delta$ is an increment

$w_{ij}$ is the weight vector.

$x$ is the input vector.

$\alpha$ is the learning rate.

An Adaline model is illustrated in Fig. 3.5; this model will be used to explain the Widrow-Hoff Delta rule.

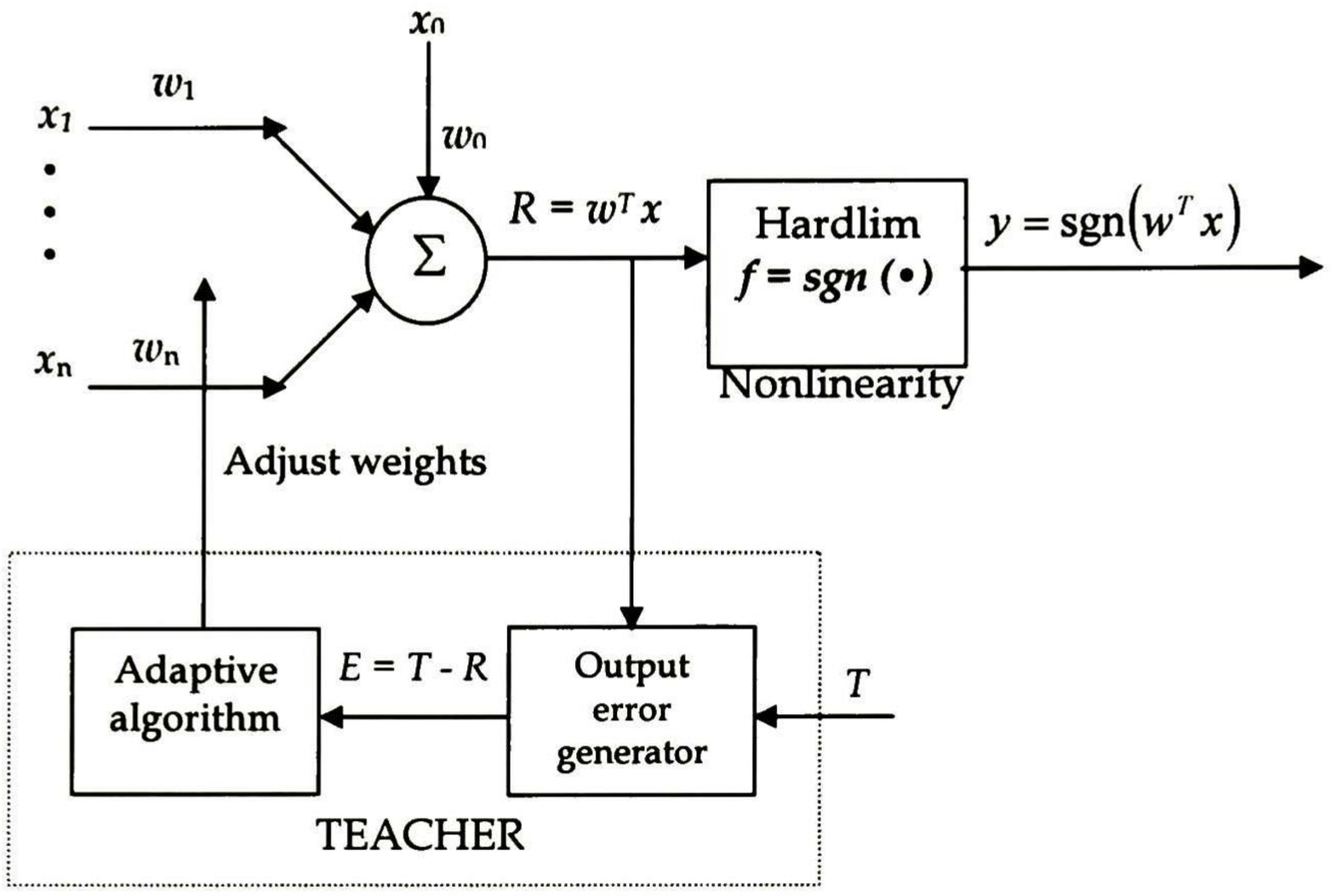



Fig. 3.5 Adaline model with Teacher.

where,

$x_j = [x_1, x_2, \ldots, x_n]$ is the input vector.

$w_j = [w_1, w_2, \ldots, w_n]$ is the input weight vector.

$R$ is the output of the neuron preceding the nonlinearity.

$O$ is the output from the neuron following nonlinearity.

$T$ is the target signal (this, along with the input signals, produces the error or learning signal used only during training).

$E$ is the output error used during learning.

Adaline Training Algorithm.

For Adaline network the learning rule which is frequently used is the Widrow-Hoff Delta rule ($\alpha$-LMS). This algorithm is based on an approximate steepest descent procedure [1, 2, 5]. The $\alpha$-LMS algorithm estimates the MSE by using the squared error at each iteration. Subsequently, the learning algorithm is described.

1. Assign random weight values.
2. Apply the selected input and the target output to the model.
3. Determine the error signal.
4. Adjust the weights based on equation (3.10) then, the error will be reduced by $1/n$, where $n$ is the number of weights.
5. Repeat the procedure expecting the error to become zero.

6. Repeat the procedure for the next set of inputs.

### 3.3.3 Recurrent Networks.

In these networks, the outputs of some neurons are feedback for the same neurons or they can also be feedback for neurons on preceding layers. Therefore, signals can run in both directions, forward and backward. A special feature of this kind of networks is their dynamic memory; it means that their outputs at a given moment reproduce the recurrent input in addition to previous inputs and outputs. Some examples of recurrent networks are the Hopfield network, the Elman network and the Jordan network.

### 3.3.4 Multi-Layer Perceptron (MLP) Network.

The arising of multi-layer perceptron (MLP) was mainly due to the limitations of the Perceptron and Adaline Networks to solve complex problems. An example of these problems is that Perceptron or Adaline networks cannot differentiate between two linearly separable sets of patterns such as the solution to the Exclusive-OR function, as it is shown in Fig. 3.6.

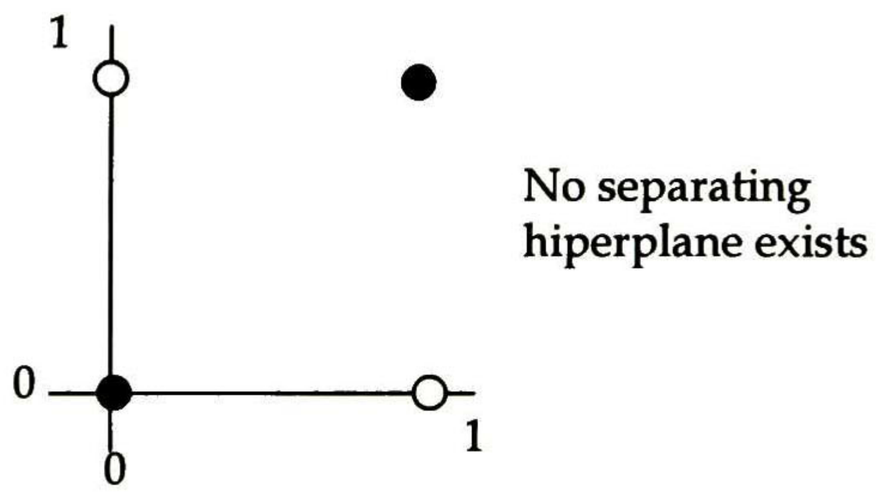



Fig. 3.6 Exclusive-OR function.

Other comment against the Perceptron and Adaline network is that for a different set of input patterns a different network has to be trained. Thus, these are some of the main reasons to develop an ANN that solve non-linear problems; this can be done connecting neural networks as Perceptron and Adaline in such a way that linear combinations are able to solve problems as an Exclusive-OR function or the square doughnut problem, Fig. 3.7.

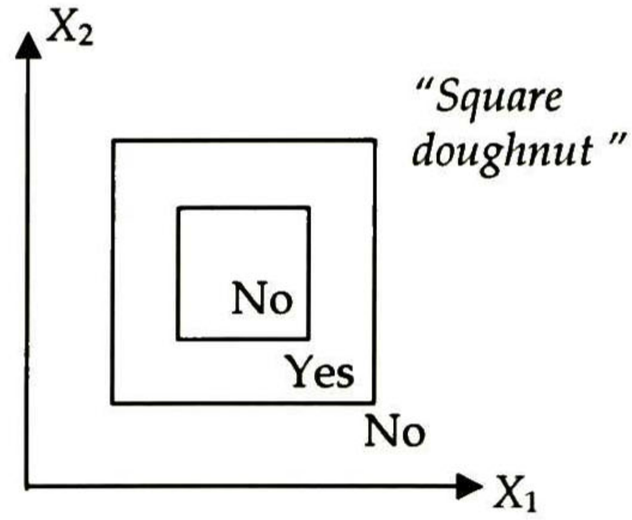



Fig. 3.7 Squared donut.

According to the limitations and the seriously limited capabilities of ANNs such as Perceptron and Adaline, the multi-layer Perceptron (MLP) was developed with the purpose to overcome these limitations. Hence, the supervised learning algorithm for Perceptron and Adaline must be renewed to

be used in MLP. The learning algorithm made for MLP is the *Back-Propagation* (BP) learning algorithm which was developed independently by Amari (1967, 1968), Bryson and Ho (1969), Werbos (1974) and Parker (1985) [1, 2, 5, 9 ,10]. Fig. 3.8 shows the BP applied to a feedforward ANN.

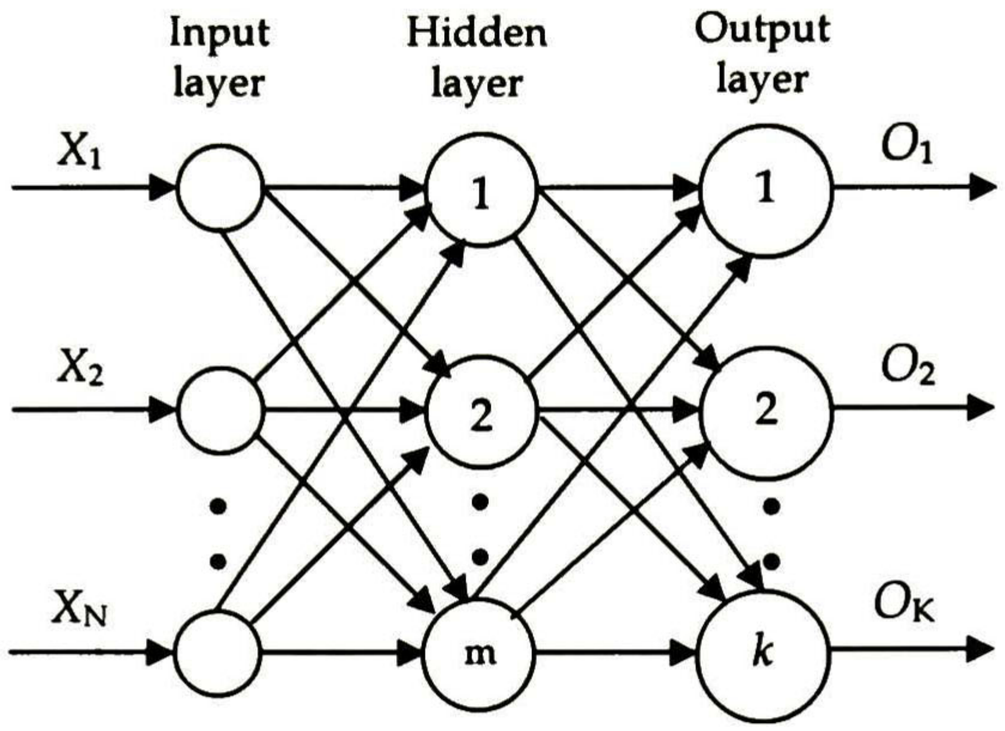



Fig. 3.8 Feedforward MLP.

There are some interesting features about the MLP structure. From Figure 3.8, each circle represents an artificial neuron. The transfer function of each one is generally the same for all the neurons. The neuron's number usually differs for each layer and it has a high dependency with respect to the problem which is going to be solved. A typical multi-layer perceptron ANN contains from 3 to 4 layers taking into account the input layer. In Table 3.1 a feedforward artificial neural network is defined with two layers of neurons.

Table 3.1 Feedforward ANN with two layers of neurons.

| Layer | Index | Input | Weights | Weighted Sum | Output | Target Output |
|---|---|---|---|---|---|---|
| Output | $k$ | $O_{nj} = s(y_{nj})$ | $w_{nj}$ | $y_{nk} = \sum_{i=0}^{D} w_{kj} O_{nj}$ | $O_{nj} = s(y_{nk})$ | $T_{nk}$ |
| Middle (hidden) | $j$ | $X_{ni}$ | $w_{ji}$ | $y_{nj} = \sum_{i=0}^{D} w_{ji} x_{ni}$ | $O_{nj} = s(y_{nj})$ | |
| Input | $i$ | $X_{ni}$ | | | | |

3.3.5 Back-Propagation Learning Algorithm.

As it is previously mentioned, the BP algorithm is one of the most frequently used supervised learning algorithms for feedforward ANNs. The aim is to adjust the weights $w_{ij}$ and $w_{ji}$ in such way that the error function is minimized over the training set. The creation of this supervised learning algorithm was a consequence of the drawbacks that the gradient-descent algorithm had demonstrated to be capable of updating the weights of the hidden layers; this was due to the hidden layers have not

available target values (desired outputs). BP is a *gradient-descent-based* learning procedure for minimizing the *sum of squared error criterion function* in a feedforward-layered network of sigmoidal units. A total procedure for updating the weights in feedforward ANNs applying the BP learning algorithm is summarized underneath, this procedure is for a two-layer architecture [5].

A. Initialise all weights, $w_{lj}^{c}$ and $w_{ji}^{c}$.

B. Set the learning rate to a small positive value.

C. Select an input pattern $x^k$ from the training set (preferably at random) and spread it through the network.

D. Use the desired target $d^k$ associated with $x^k$ and compute the output layer weight changes $\Delta w_{ij}$, applying the next equation.

$$\Delta w_{lj} = w_{lj}^{new} - w_{lj}^{c} = -\alpha \frac{\delta E}{\delta w_{lj}} = \alpha (d_l - y_l) f_0^{'}(net_l) z_j \qquad (3.14)$$

E. Employ the next equation to determine the hidden-layer weight changes $\Delta w_{ji}$.

$$\Delta w_{ji} = \alpha \left[ \sum_{l=1}^{L} (d_l - y_l) f_0^{'}(net_l w_{lj}) \right] f_h^{'}(net_j) x_i \qquad (3.15)$$

F. Update all weights according to the next equation.

$$w_{lj}^{new} = w_{lj}^{c} + \Delta w_{lj} \text{ and } w_{ji}^{new} = w_{ji}^{c} + \Delta w_{ji} \qquad (3.16)$$

for the output and hidden layers, respectively.

G. Test of convergence, which could be done by checking some pre-selected function of the output error to see if its magnitude is below some preset threshold. If convergence is met, stop; otherwise, $w_{ji}^{c} = w_{ji}^{new}$ and $w_{lj}^{c} = w_{lj}^{new}$, and go to step C. A convenient selection is the root-mean-squared (RMS) error.

where,

$l = 1, 2, \ldots, L$

$j = 0, 1, \ldots, J$

$\alpha$ is the learning rate.

$L$ is the l-th output layer.

$J$ is the j-th hidden layer.

$w_{lj}$ is the weight of the $l$-th hidden unit associated with hidden signal $z_j$.

$w_{ji}$ is the weight of the $j$-th hidden unit associated with hidden signal $x_i$.

$w_{lj}^{new}$ and $w_{lj}^{c}$ represent the updated (new) and current weight values.

$d_l$ is the $l$-th component of target vector.

$y_l$ is the $l$-th component of unit (neuron) output.

$net_l = \sum_{j=0}^{J} w_{ij} z_j$ , is the weighted sum for the $l$-th output unit.

$$z_j = f_h\left(\sum_{i=0}^{n} w_{ij} x_i\right) = f_h\left(net_j\right)$$

$f_0^{'}$ is the derivate of $f_0$ with respect to *net*.

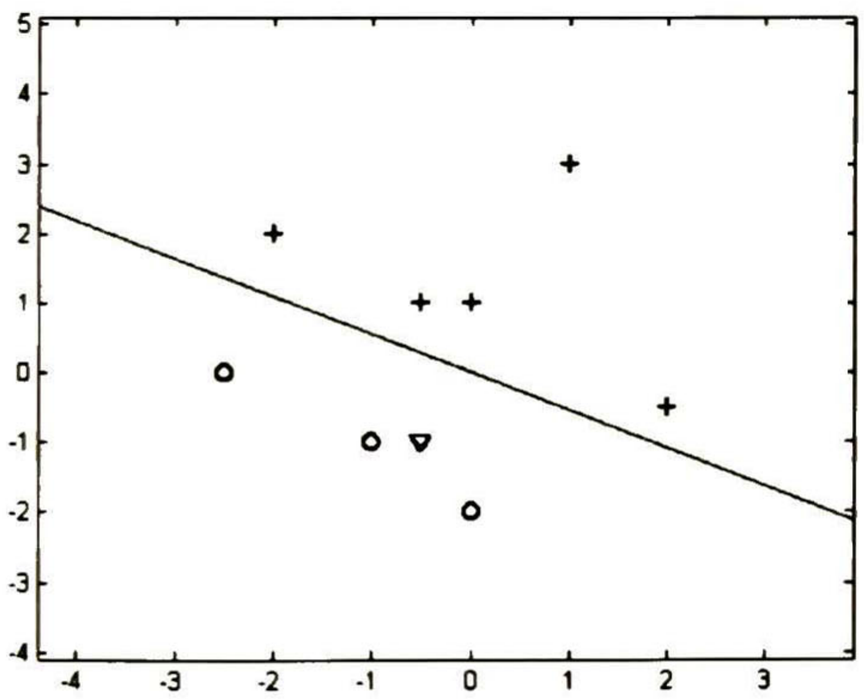
$f_0$ is the activation function for each unit of the output layer.

$f_h$ is the activation function of the hidden units, usually a hyperbolic tangent or logistic function.

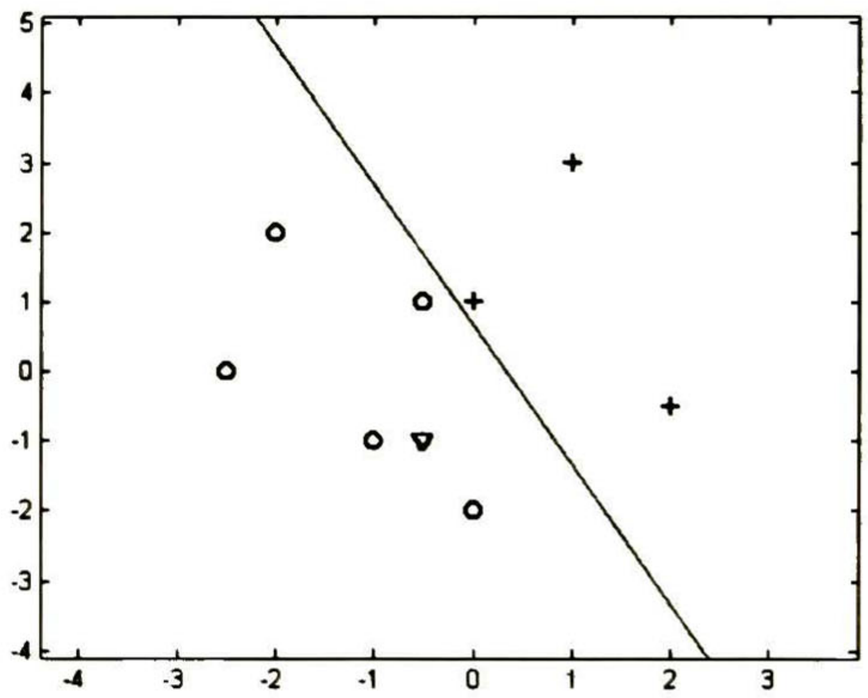
### 3.4 Some examples related to ANNs.

Ex 1. At this stride some smooth examples concerning the application of ANNs are illustrated. The first illustration (Fig. 3.9), corresponds to Perceptron networks. For this simulation, eight patterns which previously have been classified (circles or crosses) are distributed on the x - $y$ plane. The network must be robust enough to know how to divide the plane into two sections. Afterwards, another pattern is marked on the plane and the network can recognize the type of input corresponding (circle or cross). This is a peculiarity of ANNs; it could be said that the artificial neural network possesses a memory system that can envisage the output to which the new pattern corresponds, to distinguish this pattern it is marked as a triangle. This model is able to unravel many learning patterns as the user settles into two classes and another pattern can be added to the plane to be identified.

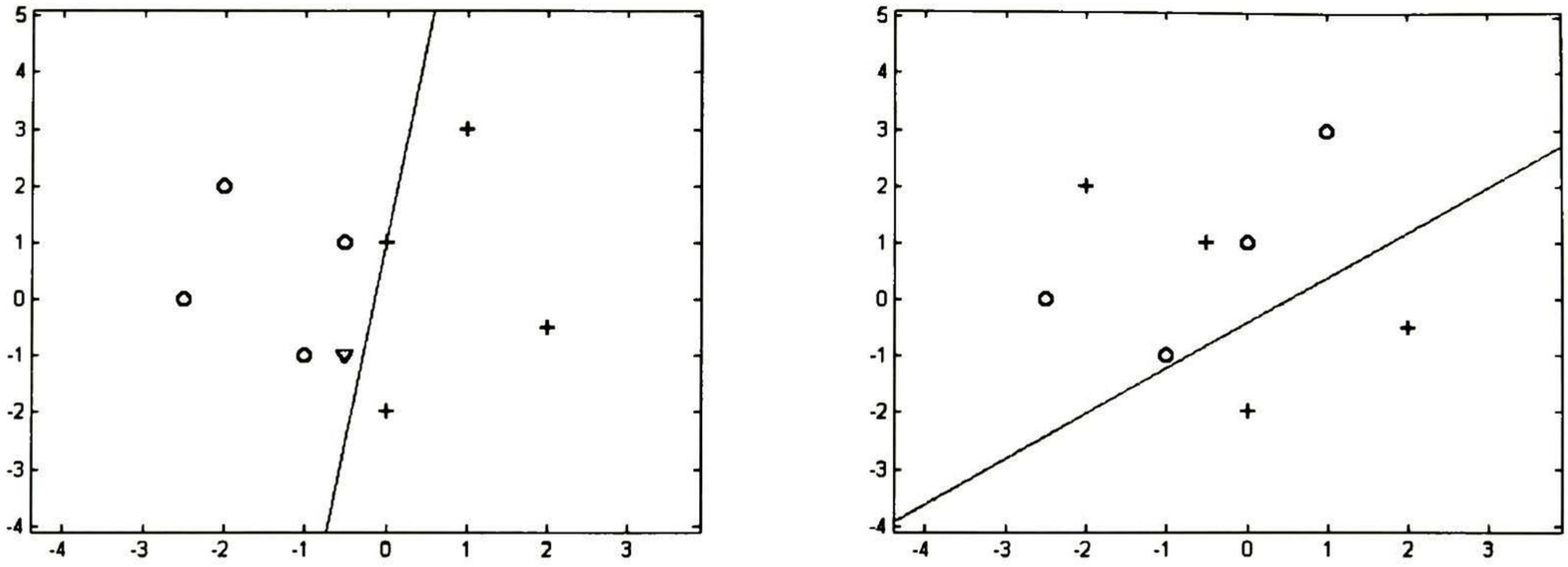
There are four cases for the same example. It is observed in Fig. 3.9 (d) that the network is not capable of finding a separable gap between them, so the pattern that will be marked after the training procedure will never appear; this is a limitation for this Perceptron network.




(a)




(b)



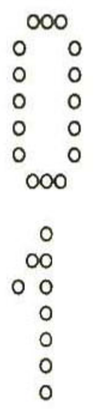
(c) (d)

Fig. 3.9 Classified vectors by a perceptron network.

Ex. 2 In the following an Adaline network is used. The application of this network will be used for pattern recognition; to be more specific the Adaline will classify digits from 0 to 9, according to some previous important information that has been settled by the programmer such as the number of layers, the number of units per layer, the activation function for each layer or unit, the learning rate, the error to finish training amongst others parameters. To identify digits, a diagonal matrix is used; the following matrix encloses data which Adaline network distinguishes. For example, if data input are [0 0 0 0 0 1 0 0 0 0] the Adaline can recognize to which number it refers to; it denotes number 5. If the next matrix represents input patterns,

$$Input = \begin{bmatrix} yes & no & no & no & no & no & no & no & no & no \\ no & yes & no & no & no & no & no & no & no & no \\ no & no & yes & no & no & no & no & no & no & no \\ no & no & no & yes & no & no & no & no & no & no \\ no & no & no & no & yes & no & no & no & no & no \\ no & no & no & no & no & yes & no & no & no & no \\ no & no & no & no & no & no & yes & no & no & no \\ no & no & no & no & no & no & no & yes & no & no \\ no & no & no & no & no & no & no & no & yes & no \\ no & no & no & no & no & no & no & no & no & yes \end{bmatrix}$$

Thus, the answer given by the Adaline network is:



If the input data arrange is changed, the results will be completely different, based on evident reasons.

Ex 3. For next case a back-propagation network with bias terms and momentum is used. *Momentum* is a simple term that helps to accelerate the algorithm convergence and it is introduced in the weight updating equation (Equation 3.10). This network will be used to predict the annual number of sunspots (time-series forecasting). To get a general vision regarding to sunspots a concisely description will be done.

Every 11 years the sun experiences a period of activity named *solar maximum*, followed by a period of calm called the solar minimum. During the solar maximum, there are many sunspots, solar flares, and coronal mass ejections, all of which can affect communications and weather here on Earth. One way to track solar activity is by observing sunspots. Sunspots are relatively cool areas that appear as dark blemishes on the face of the sun. They are formed when magnetic field lines just below the sun's surface are twisted and push through the solar photosphere. The twisted magnetic field above sunspots are sites where solar flares are observed to occur [8].

ANN will forecast the sunspots for 20 years (from 1960 to 1979). Input patterns will be annual number of sunspots for the 280 subsequent years. The main characteristics for this network are: it has three layers, in which the first one - input layer - contains thirty neurons or units, the second one has ten neurons, the hidden layer and the final layer just holds one unit. The momentum is set up at 0.5 and

the learning rate is 0.05. The transfer function used is a sigmoid function. The first weights values are chosen randomly.

The results after the network has been trained for 320 epochs are the following (Table 3.2). The MSE on training set was 0.141.

Table 3.2 Sunspot forecast.

| YEAR | SUNSPOT | PREDICTION |
|---|---|---|
| 1960 | 0.572 | 0.532 |
| 1961 | 0.327 | 0.334 |
| 1962 | 0.258 | 0.158 |
| 1963 | 0.217 | 0.156 |
| 1964 | 0.143 | 0.236 |
| 1965 | 0.164 | 0.230 |
| 1966 | 0.298 | 0.263 |
| 1967 | 0.495 | 0.454 |
| 1968 | 0.545 | 0.615 |
| 1969 | 0.544 | 0.550 |
| 1970 | 0.540 | 0.474 |
| 1971 | 0.380 | 0.455 |
| 1972 | 0.390 | 0.270 |
| 1973 | 0.260 | 0.275 |
| 1974 | 0.245 | 0.211 |
| 1975 | 0.165 | 0.181 |
| 1976 | 0.153 | 0.128 |
| 1977 | 0.215 | 0.151 |
| 1978 | 0.489 | 0.316 |
| 1979 | 0.754 | 0.622 |

Ex. 4 In the next example, a cone is formed and the ANNs will attempt to simulate it. The artificial neural network that will be employed has the next characteristics:

- ✓ It is a multi-layer feedforward network.
- ✓ It possesses one input layer with four elements.
- ✓ It includes two hidden layers each one with 20 neurons.
- ✓ A tangent-sigmoid transfer function is used for the input and hidden layers.

- ✓ A linear transfer function is applied for the single output layer, and
- ✓ The Levenberg-Marquardt algorithm is used for training the net.

Fig. 3.10 illustrates the cone reproduced by the ANN without training, and in Fig. 3.11 is presented the resultant shape of the cone after it has been trained during 200 epochs and for a MSE equal to 1e-5.

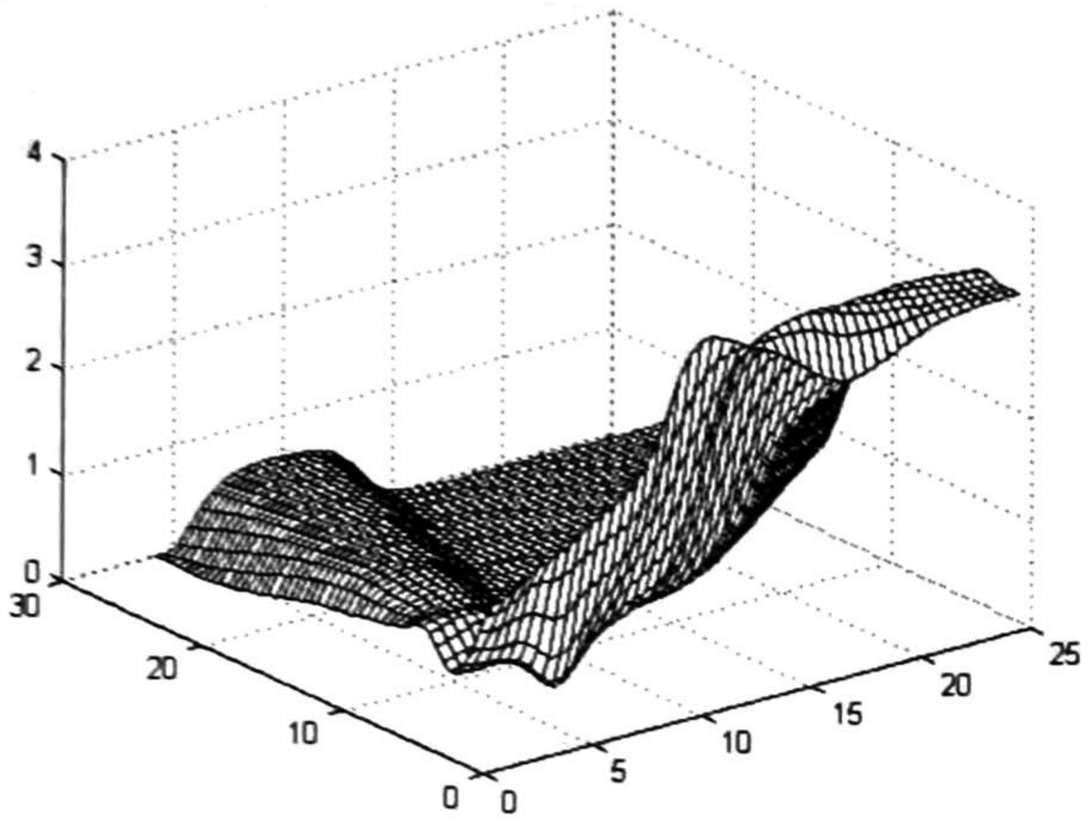



Fig. 3.10 Cone without training.

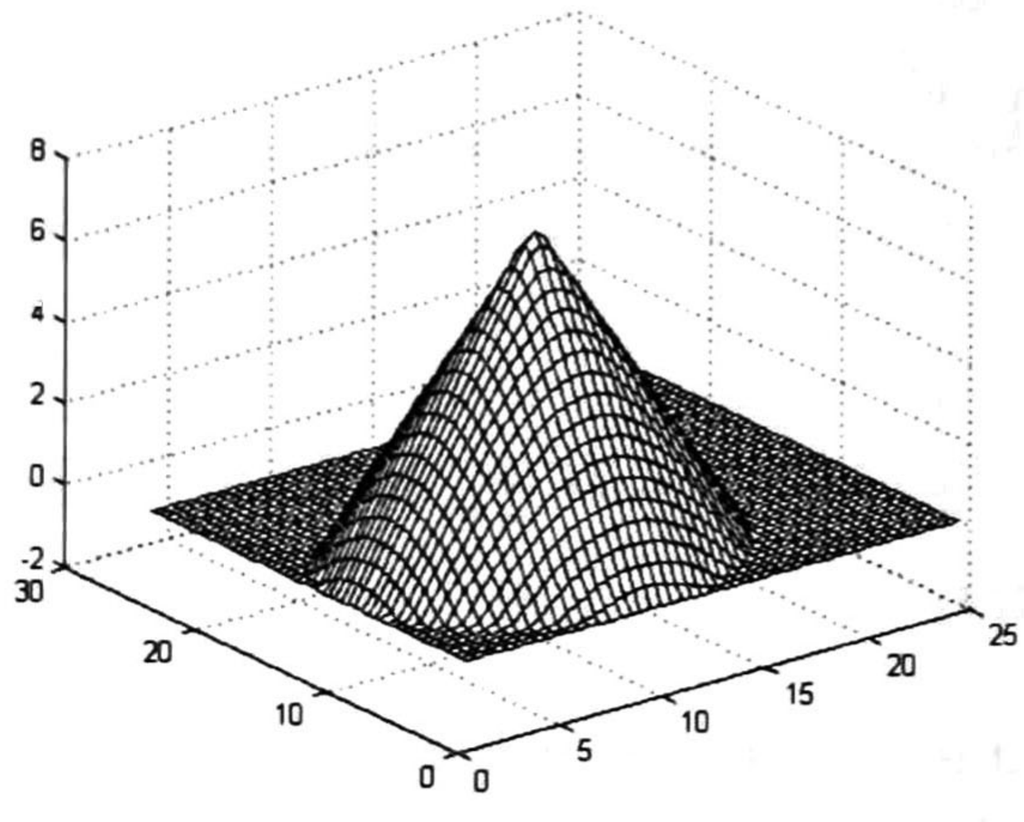
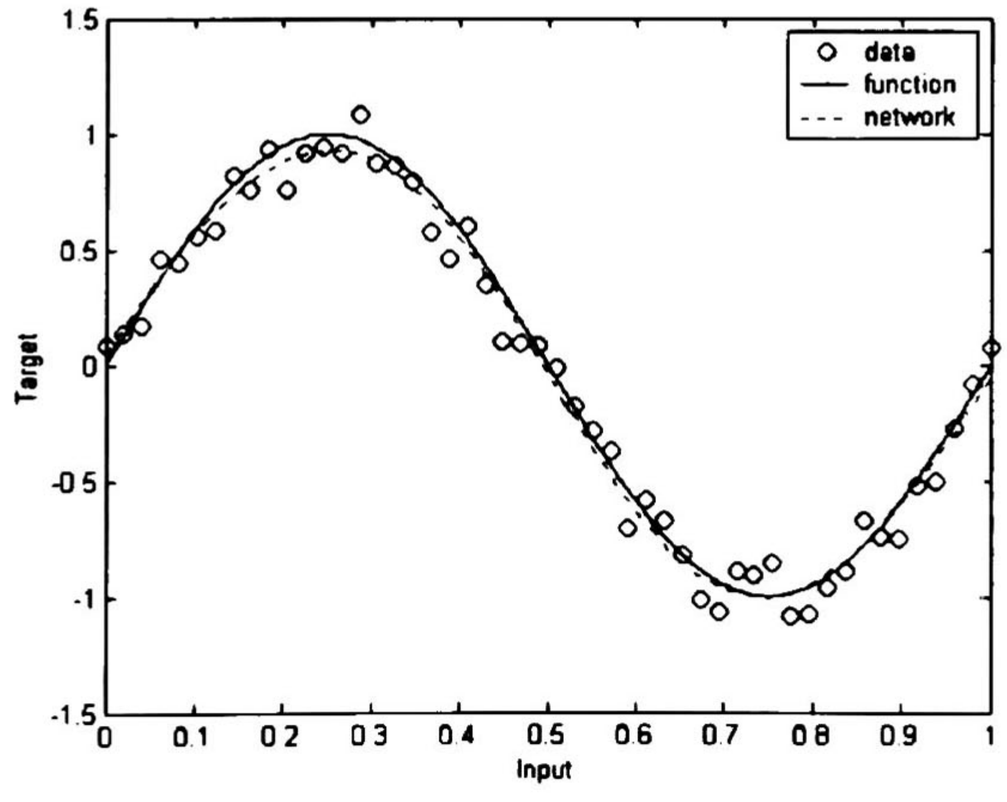


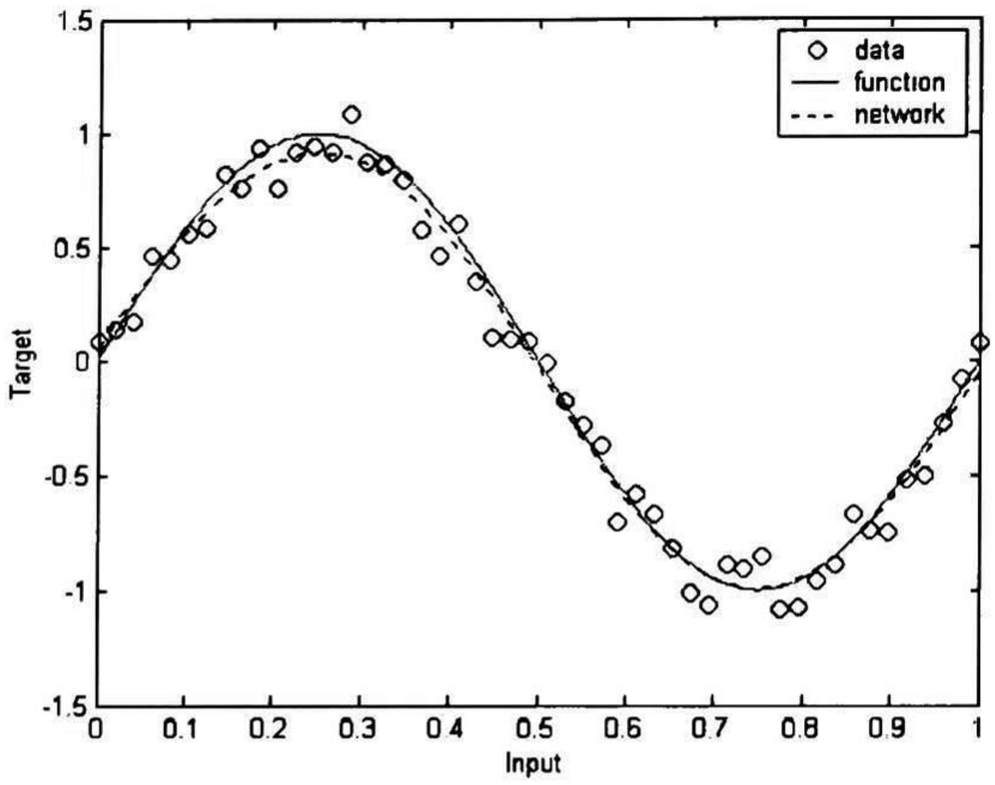

Fig. 3.11 Cone trained.

Ex 5. Other examples are employed to exemplify a multi-layer perceptron network.

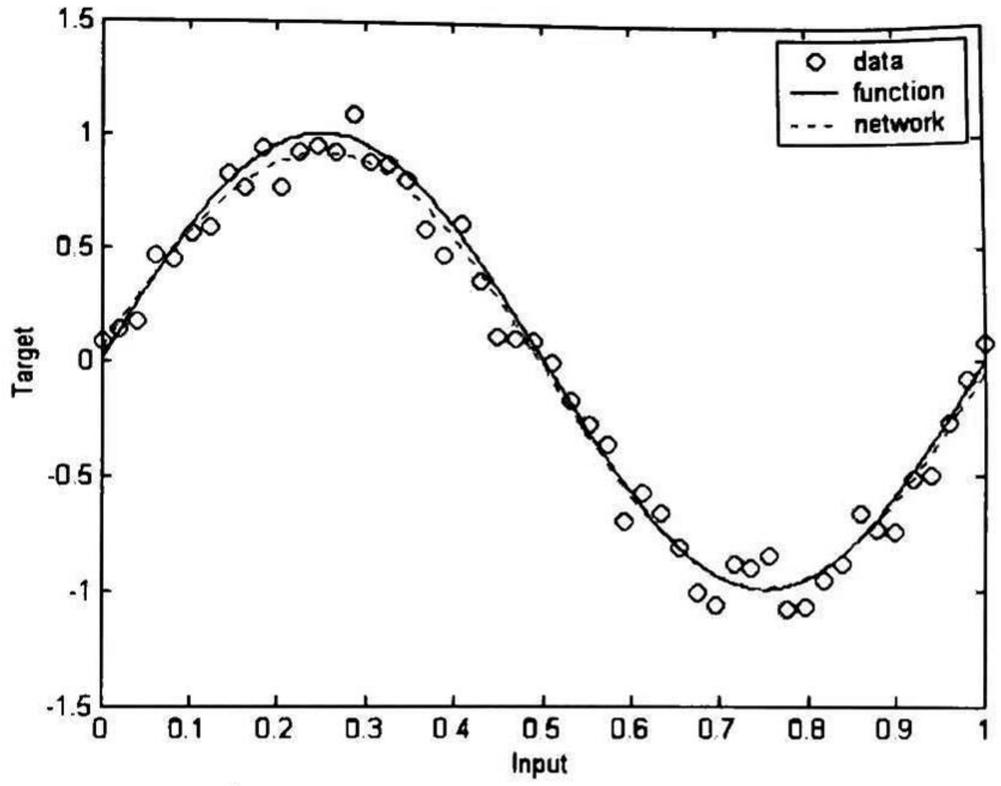
A 2-layer, feed-forward network is employed. The problem is made of one input variable X and one target variable T, with data generated by sampling X at equal intervals and then generating target data by computing sin ($2*\pi*X$) and adding Gaussian noise. The network with linear outputs is trained by minimizing a sum-of-squares error function using different optimisation algorithms. All examples have one input unit, three hidden units and one output unit, a linear output unit activation function is used, the hidden units are activated by a tangent-hyperbolic function and are trained for 500 epochs. The first example, Figure 3.12(a) is trained using a scaled conjugated gradient optimiser. Figure 3.12(b) and 3.12(c) are trained applying the quasi-Newton optimisation approach and the conjugate gradient optimisation method, respectively. All examples are trained with the same momentum and learning rate, 0.5 and 0.05 correspondingly.



(a)




(b)




(c)

Fig. 3.12 Plot results.

As it is noticed in the plot results, the ANN is competently trained for the main purpose not with standing the optimisation approach applied.

Ex. 6 The next example studies the identification of a RLC circuit that is done using an ARX (AutoRegressive, eXtra input) model structure; the RLC circuit is illustrated in Fig. 3.13.

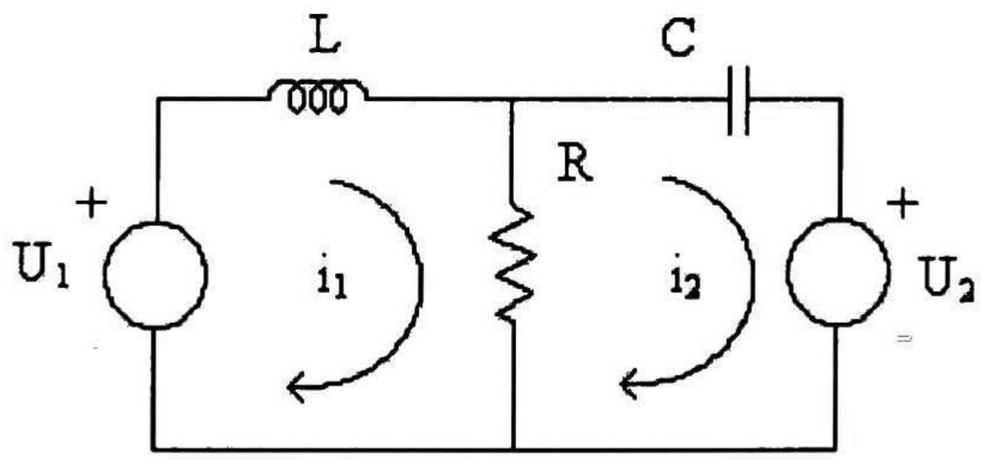



Fig. 3.13 RLC circuit.

Where $i_1(t)$ and $i_2(t)$ are the current signals, $U_1(t)$ and $U_2(t)$ are the input signals. The input signals are considered as a pseudorandom binary signal (PRBS), which is a periodic, deterministic, random process that assumes only two values with white-noise-like properties. Fig. 3.14 shows a PBRS.

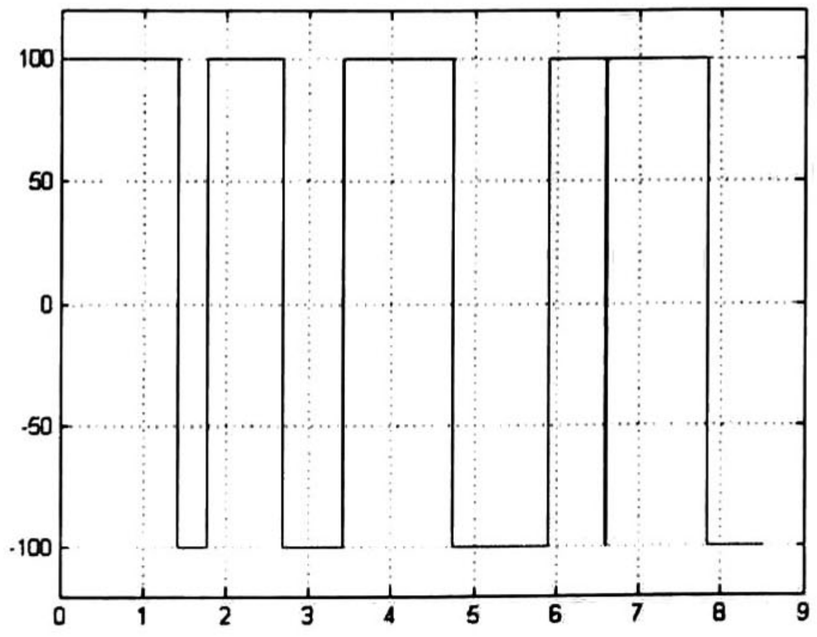


Fig. 3.14 A Pseudorandom binary signal.

The R, L, C elements are defined as

- R is the resistance value in ohms, 1.5.
- L is the inductance value in henries, 0.25.
- C is the capacitance value in faradays, 0.5.

For this RLC circuit the state space variables are the loop current, $i_1(t)$ and the voltage in the capacitance element, $V_c(t)$. The transfer functions for this RLC circuit from input 1 to output 1 and 2 are given by

$$y_1(t) = \frac{4s + 5.333}{s^2 + 1.333s + 8}$$
$$y_2(t) = \frac{4s}{s^2 + 1.333s + 8} \tag{3.17}$$

and from input 2 to output 1 and 2

$$y_1(t) = \frac{4s}{s^2 + 1.333s + 8}$$
$$y_2(t) = \frac{-0.6667s^2 - 4s}{s^2 + 1.333s + 8} \tag{3.18}$$

The ARX archetype structure is a parametric model, in which the purpose is to obtain the coefficients from the transfer functions. The parameters can be estimated applying a linear least squares technique since the predictor error is linear in the parameters. The ARX model is depicted in Fig. 3.15, and is described by

$$A(q)y(k) = B(q)u(k) + v(t) \tag{3.19}$$

where $A(q)$ and $B(q)$ represents matrix polynomials in the time operator $q$, $u(k)$ is the input vector, $y(k)$ is the output vector and $v(t)$ is the noise signal.

The optimal ARX predictor is defined by [4]

$$\hat{y}(k|k-1) = B(q)u(k) + (1 - A(q))y(k) \tag{3.20}$$

which can be written as

$$\hat{y}(k|k-1) = b_1u(k-1) + \ldots + b_mu(k-m) - a_1y(k-1) - \ldots - a_my(k-m) \tag{3.21}$$

assuming that deg(A) = deg(B) = m.

The ARX predictor presents a characteristic that makes it unique; it is always stable even if the ARX model is unstable. This distinctive feature shows because the predictor does not possess feedback.

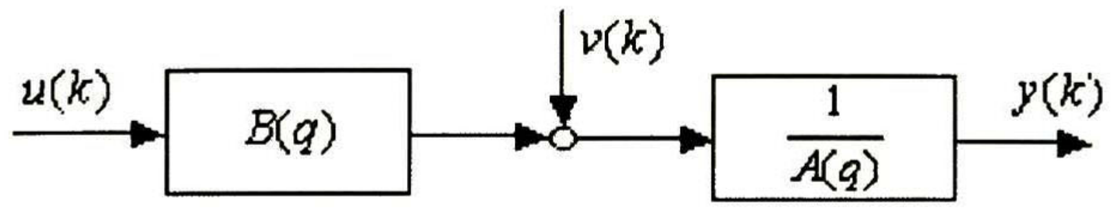



Fig. 3.15 ARX model.

An artificial neural network is trained with the objective of forecasting and obtaining the state space variable values. The ANN structure consists of two signals for the input layer, one hidden layer that has three neurons and two signals for the output layer. This ANN is a multi-layer feedforward network that is trained for 1000 epochs as maximum, using the Levenberg-Marquardt optimisation algorithm, a tangent-sigmoid transfer functions are used for the hidden layers, and linear transfer functions are applied for the output layers. All data before been applied to the ANN are scaled; this is zero mean and variance 1; this is done with the objective of not having a too dominating magnitude from the largest signal. Besides, scaling data makes the algorithm training robust and leads to a faster convergence. The ANN is trained for the first 500 samples in both inputs (PRBS) and output signal, and the next 200 samples over the state space variable values $(i_1(t), V_c(t))$ are predicted. The results are illustrated in Fig. 3.17.

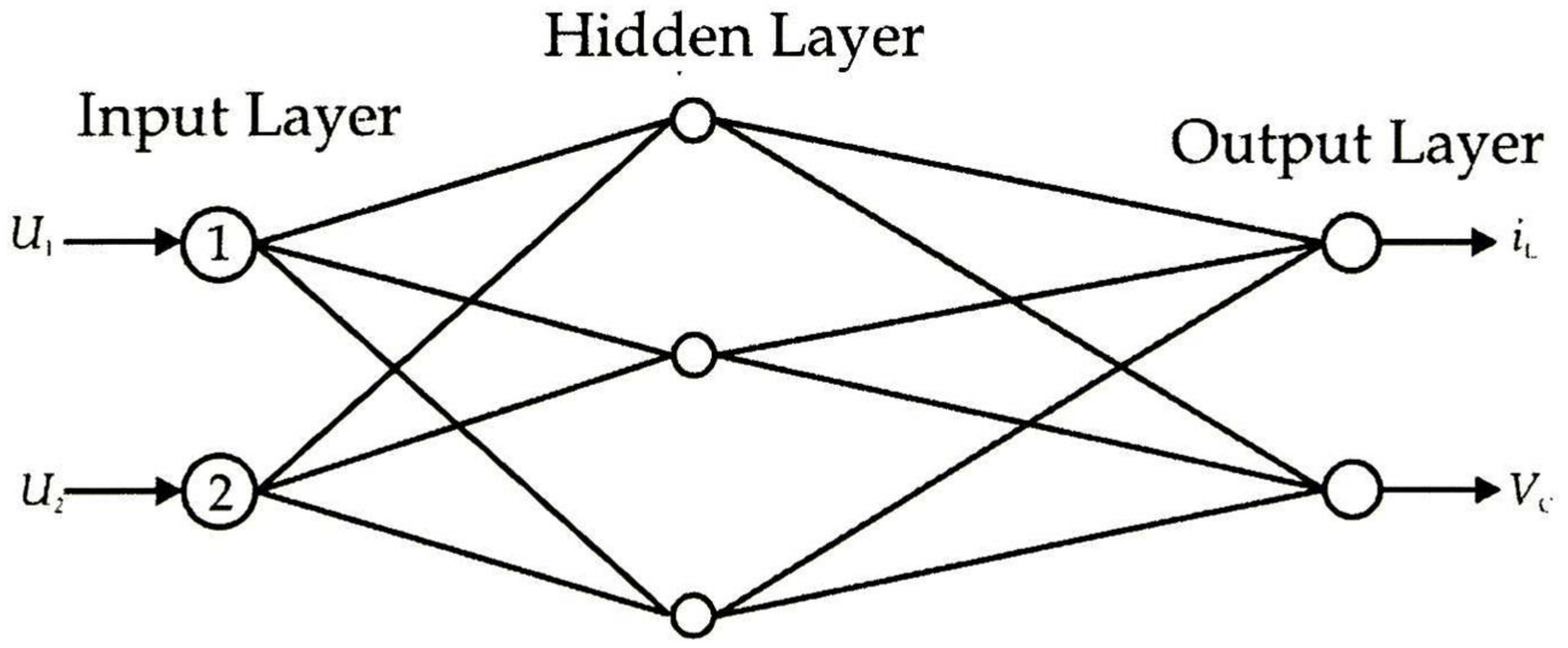



Fig. 3.16 ANN used for the RLC circuit example.

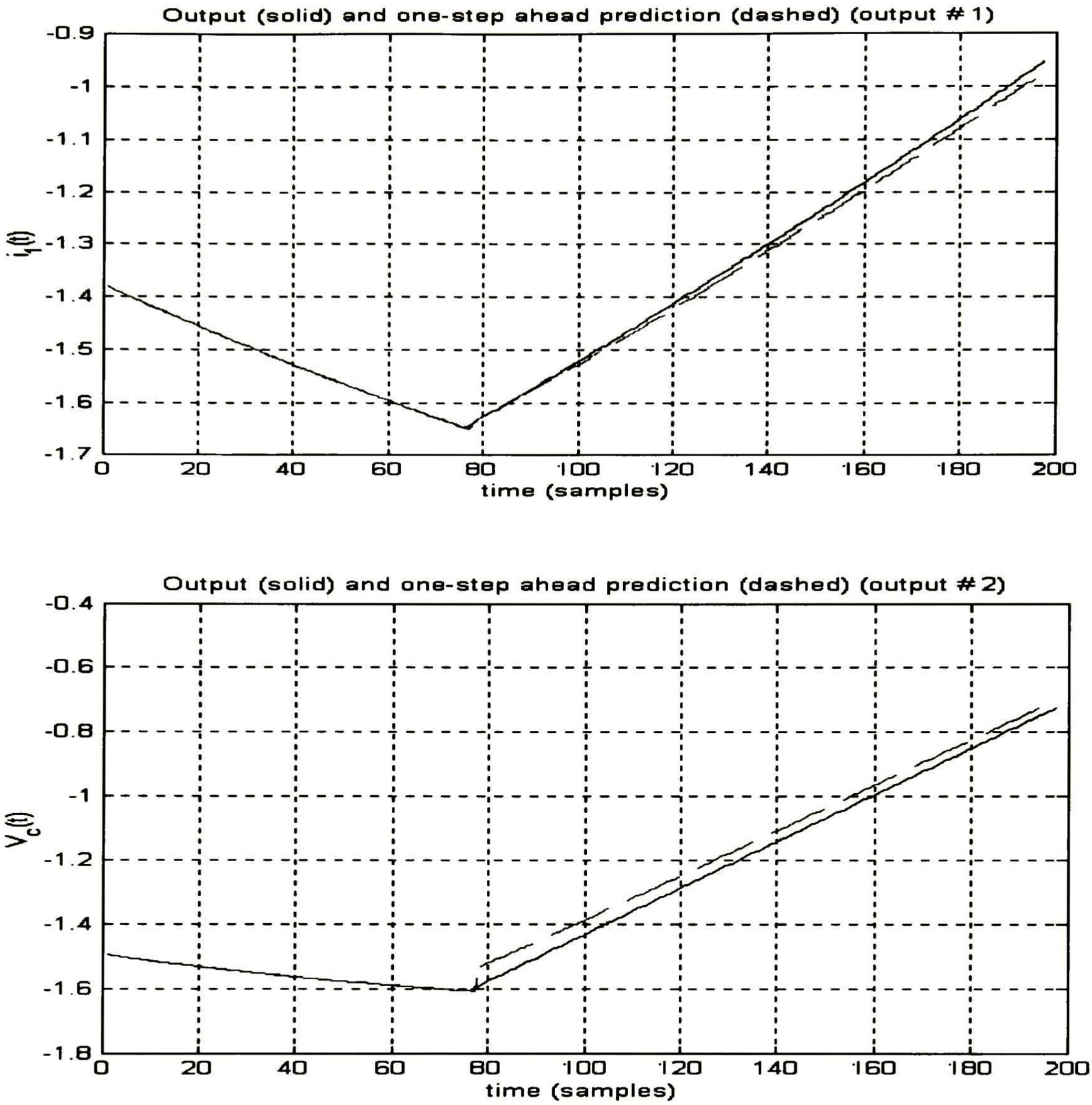



Fig. 3.17 Comparison between output and prediction values.

## SYNOPSIS.

In this chapter the foundations of artificial neural networks are reviewed, the mainly universal category of them such as Perceptron, Adaline and MLP and their corresponding training algorithms are illustrated. It is confirmed that the best ANN that use supervised learning is the multi-layer perceptron network due to its capability for solving all the problems and sundry them where they were showed off. Perceptron and Adaline, primarily the first one, presented trouble for solving the Ex.1 described, so its limited capabilities are corroborated. It is also surveyed the principal optimiser algorithms apply to artificial neural networks and their principal advantages and disadvantages concerning each other. An example application for a RLC circuit is presented and it is the beginning for the next chapter, where more applications are described.

## REFERENCES


1. Kartalopoulos V. Stamatios, *Understanding Neural Networks and Fuzzy Logic*, IEEE Press, 1996.
2. Müller Berndt, Reinhardt J, Strickland M. T., *Neural Networks: An Introduction*, Springer-Verlag, 1995.
3. Rao Singiresu S., Engineering Optimization, John Wiley & Sons, 1996.
4. Norgaard M, et al, *Neural Networks for Modelling and Control of Dynamic Systems*, Springer-Verlag, 2000.
5. Haykin Simon, *Neural Networks A Comprehensive Foundation*, Prentice-Hall, 1999.
6. Mehrotra Kishan, et. al, *Elements of Artificial Neural Networks*, The MIT Press, 1997.
7. Mohamand H. Hassoun, *Fundamentals of Artificial Neural Networks*, The MIT Press, 1995.
8. Website http://science.nasa.gov.
9. Fausset Laurence, *Fundamentals of Neural Networks*, Prentice-Hall, 1994.
10. Golden M. Richard, *Mathematical Methods for Neural Network Analysis and Design*, The MIT Press, 1996.
11. S. Kirkpatrick, Jr. C. D. Gelatt and Vecchi M.P, *Optimization by Simulated Annealing*, Science, 220 (4598) pp 671-680.
12. Metropolis, A. H., et. al, *Equation of State Calculations by Fast Computing Machines*, Journal of Chem. Phys., 21, 1953, pp. 1087-1092.
13. J. J. Moré, *The Levenberg-Marquardt algorithm: implementation and theory*. Watson GA, editor. Numerical analysis, 630. Berlin: Springer, 1977. pp. 105.
14. J. H. Holland, *Outline for a logical theory of adaptive systems*. Journal of the Association for Computing Machinery, 9:297-314, 1962.
15. D. E. Goldberg. *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley, 1989.
16. M. Gen and R. Cheng. *Genetic algorithms & Engineering Design*. John Wiley & Sons, Inc., 1997

# Chapter 4

## MODAL AND NEURAL DYNAMIC EQUIVALENTS

*Imagination is more important than knowledge.*

*A. Einstein*

### 4.1 Introduction.

Researchers have made many proposals to construct dynamic equivalents based on identification, modal analysis, empirical simplification, coherency or using singular perturbation theory just to remark some of them. Actually, these methodologies are well known and have been analysed by many people. The major techniques that have been most contemplated to solve dynamic equivalents are: modal analysis, coherency and identification [1-14].

The central point of modal techniques is to represent sections of an unwieldy scale power system model by equations which are easier to solve and have the peculiarity to be more computationally manageable than the regular nonlinear differential/algebraic equations for transient stability studies. Such approaches are based on a linearized version of the power system dynamic model, understanding that the total dynamic response is composed of elemental blocks, labelled as natural modes.

The coherency procedure is applied to generator buses for reducing their number. The purpose is to evaluate the best way in which those groups can be recognized. Generally, if the angular differences of two generator buses are invariant over a certain period, with a predefined tolerance, these generator buses are recognized as coherent. Normally, coherency measures are used to this intention, and those that have offered high-quality results for dynamic equivalents are based on the internal voltage angle deviation.

The identification technique relies on the determination of the essential characteristics of a dynamic system by monitoring the response of system variables to random system inputs, either natural or intentional. This technique is different from the others due to information of the external system is not required so this could be seen as an advantage. The fundamental nature of identification approaches consists of matching signals from an actual system that is under random disturbances, with the same signals calculated on a reduced model of the system, and adjusting this one to reduce the difference. For many years such approaches have been an important guidance for many authors so that many other techniques, which are related to these ones have emerged to determine the problem such as neural networks [17-20] and statistical approaches [15, 16].

In the following, a methodology based on modal analysis is described and proposed for reducing the external system to a few generator nodes. The reduced model obtained, preserves the modal structure of the study system.

In this chapter two propositions are made to construct robust dynamic equivalents: (a) based on preserving those electromechanical modes related with the studied system; (b) based on forecasting voltages at frontier nodes through neural networks.
Robustness in this context means that the Dynamic Equivalent is able to reproduce, as close as possible, results of the full system in face of different operating conditions and faults location.

## 4.2 Modal Equivalent.

The dynamic equivalents main objective is to reduce the external system, preserving only the *frontier nodes*. That is, those nodes of the external system linked with nodes of the system studied. At frontier nodes, fictitious generators are located. Constraints are related to the loss of minimum information on electromechanical modes of the system studied [5].

### 4.2.1 Steady state.

First, the preservation of the steady state must be verified. In this work the option of eliminating all nodes of the external system, except the frontier nodes, was taken. The complex power that should inject the fictitious generators at such nodes is calculated by a load flow study. Therefore, the nodal balance equation yields

$$\sum_{j \in J} P_{ij} + Pg_i + Pl_i = 0, \forall i \in I \tag{4.1}$$

where $I$ is the set of frontier nodes, $J$ is the set of nodes linked to the $i_{th}$ frontier node, $P_{ij}$ is the active power flowing from $i_{th}$ to $j_{th}$ node, $Pg_i$ is the generation at the $i_{th}$ node and $Pl_i$ is the load at the $i_{th}$ node. The voltages of these nodes are adjusted to those values calculated in the steady state study. This procedure has the advantage of avoiding the aggregation of generators, as in the classical equivalency methods, Section 2.7.1.

### 4.2.2 Proposition.

The model of the equivalent generators located at the frontier nodes can be of any order. For the sake of simplicity, a fourth order model is applied

$$\begin{aligned}
\frac{d\delta}{dt} &= \omega - \omega_0 \\
\frac{d\omega}{dt} &= \frac{1}{T_j}\left[T_m - T_e - D(\omega - \omega_0)\right] \\
\frac{dE'_q}{dt} &= \frac{1}{T'_{d0}}\left[-E'_q - (x_d - x'_d)i_d + E_{fd}\right] \\
\frac{dE'_d}{dt} &= \frac{1}{T'_{q0}}\left[-E'_d - (x_q - x'_q)i_q\right] \\
\frac{dE_{fd}}{dt} &= \frac{1}{T_A}\left[-E_{fd} + K_A(V_{ref} + V_s - V_t)\right]
\end{aligned} \tag{4.2}$$

where $\delta$ is the power angle position, $\omega$ is the angular speed, $H$ is the inertia constant, $E_q^{'}$ and $E_d^{'}$ are the transient electro-motive forces (emfs), $I_d$ and $I_q$ are the *d*-axis and *q*-axis armature currents, $E_{fd}$ is the excitation voltage, $T'_{d0}$ is the d-axis open-circuit time constant, $T'_{q0}$ is the q-axis open-circuit time constant, $X_d$ is the d-axis synchronous reactance, $X'_d$ is the d-axis transient reactance, $X_q$ is the q-axis synchronous reactance and $X'_q$ is the q-axis transient reactance, $K_A$ is the gain and $T_A$ is the time constant of the static excitation system.

As can be noticed, a static excitation system is included. The model parameters to be estimated are: *(a)* inertia constant $H_{eqi}$; *(b)* steady state and transient reactances $X_{deqi}, X'_{deqi}, X_{qeqi}, X'_{qeqi}$; *(c)* damping factor $D_{eqi}$; *(d)* time constants, $T'_{d0eqi}, T'_{q0eqi}, T_a$; (*e*) gains $K_a$ So that, ten parameters are associated with each equivalent generator. They are estimated through an optimisation procedure. The full multi-machine power system model linearized around the $k_{th}$ equilibrium point is represented by the state space equation

$$\dot{x} = A_k x + B_k u,\ y = C_k x \tag{4.3}$$

where: $\dot{x} \in \mathbf{R}^n$ is the state space vector, $u \in \mathbf{R}^q$ is the input signal vector, and $x \in \mathbf{R}^p$ is the output signal vector. Let us define $A_{r\_k}$ as the state space matrix of the corresponding linearized reduced model around the $k_{th}$ equilibrium point, when just generators of the studied system and some fictitious generators -representing the external or reduced subsystem- are retained. Therefore, in this work the dynamic equivalent is calculated solving the following minimization problem

$$\min \sum_{k \in K} \left\{ \lambda(\mathrm{A_k}) - \lambda(\mathrm{A_{r\_k}}) \right\} \tag{4.4}$$

where $K$ is the set of operating conditions under study; $\lambda(A_k)$ is the set of electromechanical modes with relevant contributions of generators of the studied system; this set is evaluated just once for each operating condition. $\lambda(A_{r\_k})$ is the associated set of electromechanical modes of the reduced system; this set is evaluated each time is required by the optimisation algorithm, in order to adjust the equivalent generators' unknown parameters. That is, the proposition is based on matching, as close as possible, the set of electromechanical modes related with the studied system evaluated for the full system on different operating conditions $\lambda(A_k)$, with the corresponding set of electromechanical modes for the reduced system through the minimisation process, the unknown parameters of the equivalent generators are estimated, until Eq. (4.4) reaches a minimum.

### 4.2.3 Levenberg - Marquardt Algorithm.

To solve the problem posed in Eq. (4.4) several methods can be tried, but only two of them are considered appropriate in this work: (*a*) the Levenberg-Marquardt Algorithm; (*b*) Genetic Algorithms. Within the conventional optimisation methods, the Levenberg-Marquardt method was preferred due to its robustness, although another one may be useful. The optimisation problem can be expressed as a sum of squares

$$J(x) = [r(x)]^T [r(x)] \tag{4.5}$$

where $x$ is the vector of unknown parameters. To minimize $J(x)$ it is necessary to differentiate (4.5) and equate to zero, it must satisfy the non-linear equation

$$\left.\frac{dJ(x)}{dx}\right|_{x=\hat{x}} = -2[F(\hat{x})]^T r(\hat{x}) = 0 \tag{4.6}$$

where $F(\hat{x}) = \left.\frac{dr(x)}{dx}\right|_{x=\hat{x}}$ is the Jacobian matrix.

One method of solving (4.6) is based on the Taylor series approximation of $r(x)$ around a nominal point $x^0$, i.e.

$$r(x) = r(x^0) + F(x^0)[\hat{x} - x^0] \tag{4.7}$$

Substituting (4.7) into (4.6) it yields

$$\begin{aligned} [F^T(x^0)F(x^0)][\hat{x} - x^0] &= F^T(x^0)r(x^0) \\ [F^T(x^q)F(x^q)][\Delta x^{q+1}] &= F^T(x^q)r(x^q) \end{aligned} \tag{4.8}$$

with update values $x^{q+1} = x^q + \Delta x^{q+1}$ The iterations of (4.8) are continued until $J(\hat{x})$ approaches the minimum. The method of estimating $\hat{x}$ by solving (4.8) is also called the Gauss-Newton method. According to the Levenberg-Marquardt algorithm [22], (4.8) may be solved by adding positive numbers to the diagonal of the matrix $F^T(x^q)F(x^q)$ in case of oscillatory behaviour in convergence and/or ill-conditioning of the matrix. So, (4.8) becomes

$$[F^T(x^q)F(x^q) + \alpha D]\Delta x^{q+1} = F^T(x^q)r(x^q) \tag{4.9}$$

where D is a diagonal matrix and the constant $\alpha > 0$. A small $\alpha$ gives a Newton's step and a large $\alpha$ gives a steepest descent step. We adjust $\alpha$ by comparing the actual reduction $\Delta J(\hat{x})$ in the sum of squares, to the reduction that would have occurred if the linear model

$$r(x^0 + \Delta x) = r(x^0) + F(x^0)\Delta x \tag{4.10}$$

had been precise.

A test for optimality of the point $x^q$ often carried out is: if $\left\| \frac{dJ(x^q)}{dx} \right\| \leq \varepsilon$, $x^q$ is optimum and hence stops the process. Smaller convergence values (ε) result with the best estimation of the model parameters' model.

### 4.2.4 Genetic Algorithm (GA)

Genetic algorithms are global, randomised search techniques based on the mechanics of natural selection and natural genetics [23]. They were developed to allow computers to evolve solutions to difficult problems, such as function optimisation and artificial intelligence. In a GA, solutions represented by data structures called *individuals* are evolved, and new population of individuals are created. Every individual is assigned a *fitness* measure that characterizes how it compares to other individuals in the same population. In general, the fittest individuals of any population tend to reproduce and survive to the next generation, thus improving successive generations. However, inferior individuals can, by chance, survive and also reproduce. Genetic algorithms have been shown to solve linear and non-linear problems by exploring all regions of the state space and exponentially exploiting promising areas through mutation, crossover, and selection operations applied to individuals in the population [24]. During the course of an algorithm run, population with improved solutions are evolved until a stopping criterion is met. Algorithms for function optimisation are generally limited to convex regular functions. However, many functions are multi-modal, discontinuous, and non-differentiable. Stochastic sampling methods have been used to optimise these functions. Whereas traditional search techniques use characteristics of the problem to determine the next sampling point (e.g., gradients, Hessians, linearity and continuity), stochastic search techniques make no such assumptions. Instead, the next sampled point(s) is (are) determined based on stochastic sampling/decision rules rather than a set of deterministic decision rules. Genetic algorithms have been used to solve difficult problems with objective functions that do not possess *nice* properties such as continuity, differentiability, satisfaction of the Lipschitz condition, etc. [25]. Selection procedure may create a new population for the next generation based on either all parents and offspring or part of them. A sample space is characterized by two factors: size and ingredient (parent or offspring). The *regular sampling space* contains all offspring but just part of parents. The *enlarged sampling space* contains whole of parents and offspring. Sampling mechanism concerns the problem of how to select individuals from sampling space. Three basic approaches have been used to sampling individuals: stochastic, deterministic and mixed sampling. *Selection probability* concerns how to determine selection probability for each individual. In proportional selection procedure, the selection probability of an individual is proportional to its fitness. This simple scheme exhibits some undesirable properties. *Scaling* and *ranking mechanisms* are proposed to mitigate these problems. Genetic algorithms have

proved to be a versatile and effective approach for solving optimisation problems. Nevertheless, there are many situations in which the simple genetic algorithm does not perform particularly well, and various methods of *hybridisation* have been proposed. For many optimisation problems there are multiple, equal, or unequal optimal solutions. A simple GA cannot maintain stable populations at different optimal of such functions. In case of optimal solutions with equal fitness, sampling errors of GA operators cause the population to converge to a single solution. However, in the case of unequal optimal solutions, the population invariably converges to the global optimum. A simple GA with no *niching* will converge to a single optimum. Whereas a modification of the GA process with *niching* helps in maintaining subpopulation near the global and the local optimal.
The availability of alternate solutions is of great practical utility. To achieve this objective, it is essential to introduce a controlled competition among different solutions near every locally optimal region. This would maintain stable subpopulation at such optimal regions. This could be accomplished by incorporating the concepts of *niche* and *species* into the GA search process. For the optimisation of the objective (4.4), a real-valued alphabet was employed in conjunction with the selection, mutation and crossover operators. Initialisation of the population was done by generating random strings with the search space.

### 4.2.5 Example.

A testing system having 68 nodes, 16 generators and 86 lines is used to show the applicability of the proposed methodology, Fig. 4.1 [21].

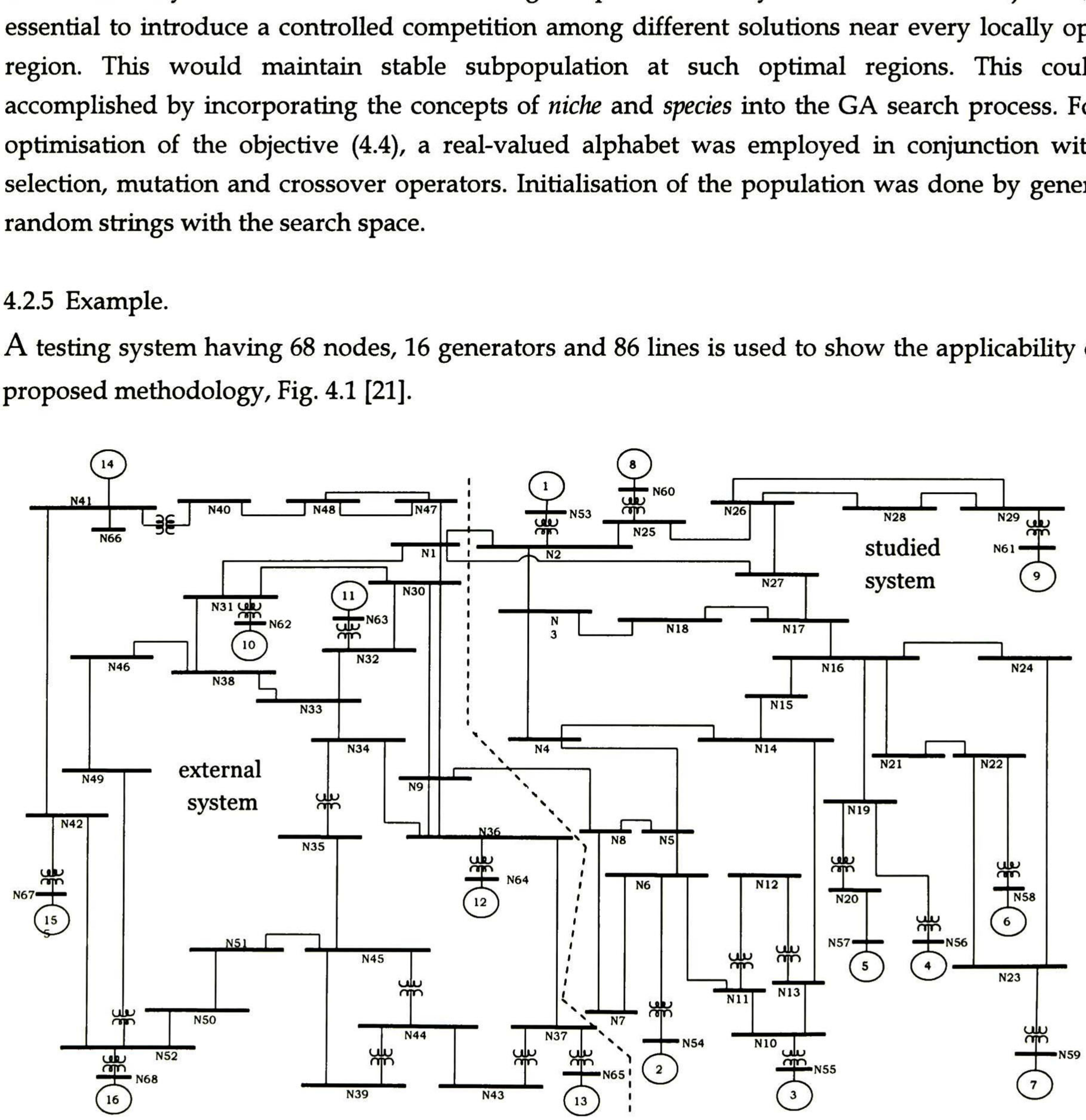



Fig. 4.1 16-machine power system.

The subsystem on the right side of the dotted line is considered as the system under study. Consequently, the subsystem on the left side is the external system. So, there are two frontier nodes (1 and 9) and three frontier lines (1-2, 1-27 and 9-8). Fig. 4.2 depicts an equivalent network including two fictitious generators at nodes 1 and 9. Models including transient effects on d and q axes are considered for all generators, which are equipped with static excitation systems, Eq. (4.2). For simplicity, the gains $Ka = 50$ and the time constants $Ta = 0.02$. Thus, the models' dimension is $n = 80$.

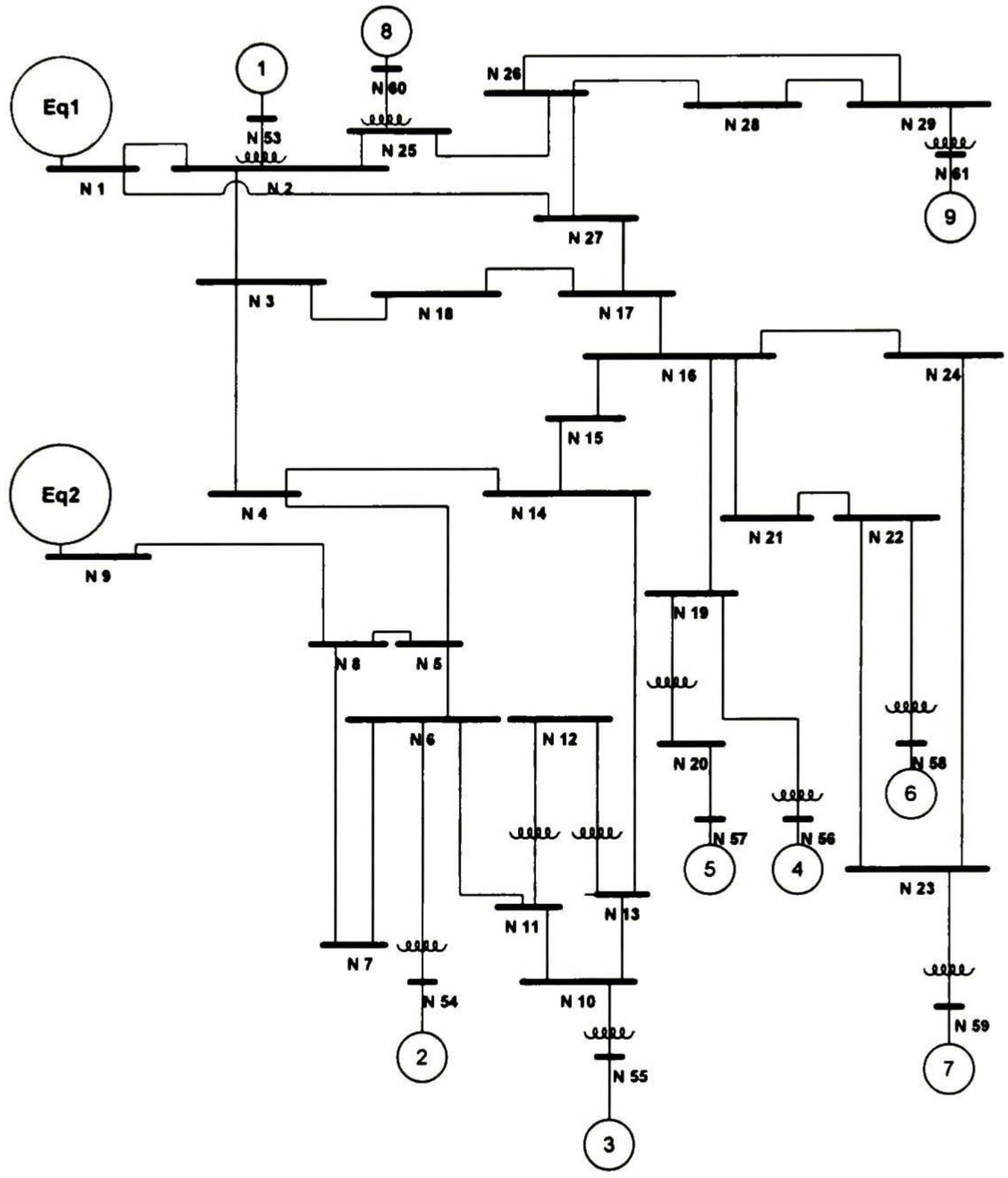



Fig. 4.2 Reduced model.

To design robust dynamic equivalents, three operating conditions are taken into account. As an example: *(a)* case 1, according to [21]. *(b)* Transmission lines 3-18 and 25-26 are out of service, *case 2*. *(c)* Transmission lines 4-14, 16-17, and 25-26 are out of service, *case 3*.

Table 4.1 shows the main electromechanical modes for the full system associated with the three operating cases taken as example. Employing the Levenberg-Marquardt algorithm, Table 4.2 exhibits the estimated parameters for the fictitious generators, under the afore mentioned considerations.

Table 4.1 Main modes associated with the study system under three operating cases.

| Case 1 | Case 2 | Case 3 |
|---|---|---|
| -0.6328 ± j 9.7121 | -0.0578 ± j 6.1957 | -0.1689 ± j 5.2650 |
| -0.6552 ± j 9.6514 | -0.1433 ± j 6.5962 | -0.1019 ± j 6.6916 |
| -0.7552 ± j 9.5476 | -0.2555 ± j 7.4775 | -0.2631 ± j 7.5026 |
| -0.3968 ± j 8.4562 | -0.4817 ± 7.6959 | -0.4905 ± j 7.6875 |
| -0.2632 ± j 8.0498 | -0.2724 ± j 8.0581 | -0.2938 ± j 8.0800 |
| -0.3108 ± j 8.0446 | -0.3283 ± j 8.3308 | -0.3338 ± j 8.3386 |
| -0.4764 ± j 7.6989 | -0.7795 ± j 9.5381 | -0.8119 ± j 9.5309 |
| -0.0956 ± j 6.9659 | -0.6498 ± j 9.6555 | -0.6515 ± j 9.6569 |
| -0.0826 ± j 6.5801 | -0.6371 ± j 9.7269 | -0.6464 ± 9.7301 |

Table 4.2 Estimated parameters for the equivalent generators

| Generator Eq. 1 | Generator Eq. 2 |
|---|---|
| Xd_1 = 0.1260 | Xd_2 = 0.1394 |
| X'd_1 = 0.1063 | X'd_2 = 0.0001 |
| T'd0_1 = 5.0132 | T'd0_2 = 5.0149 |
| Xq_1 = 0.0775 | Xq_2 = 0.1014 |
| X'q_1 = 0.0627 | X'q_2 = 0.0285 |
| T'q0_1 = 5.0143 | T'q0_2 = 5.0094 |
| Ka_1 = 66.6833 | Ka_2 = 66.6833 |
| H1 =166.75 | H2 = 166.75 |
| D1 = 0.6667 | D2 = 0.6668 |
| Ta_1 = 0.3334 | Ta_2 = 0.3334 |

Fig. 4.3-4.4 depict a sample of angles, velocities, electrical power and voltages, comparing the behaviour of the full and reduced system after a three-phase fault. To compare signals, the following RMS difference is employed

$$Error = \sqrt{\frac{1}{T}\int_0^T \{S_{full} - S_{equiv}\}^2 dt} \tag{4.11}$$

where $S$ means any signal. For instance, Table 4.3 presents the angle, velocity and electrical power errors for a three-phase fault at node 19 and the operating condition *case 3*.

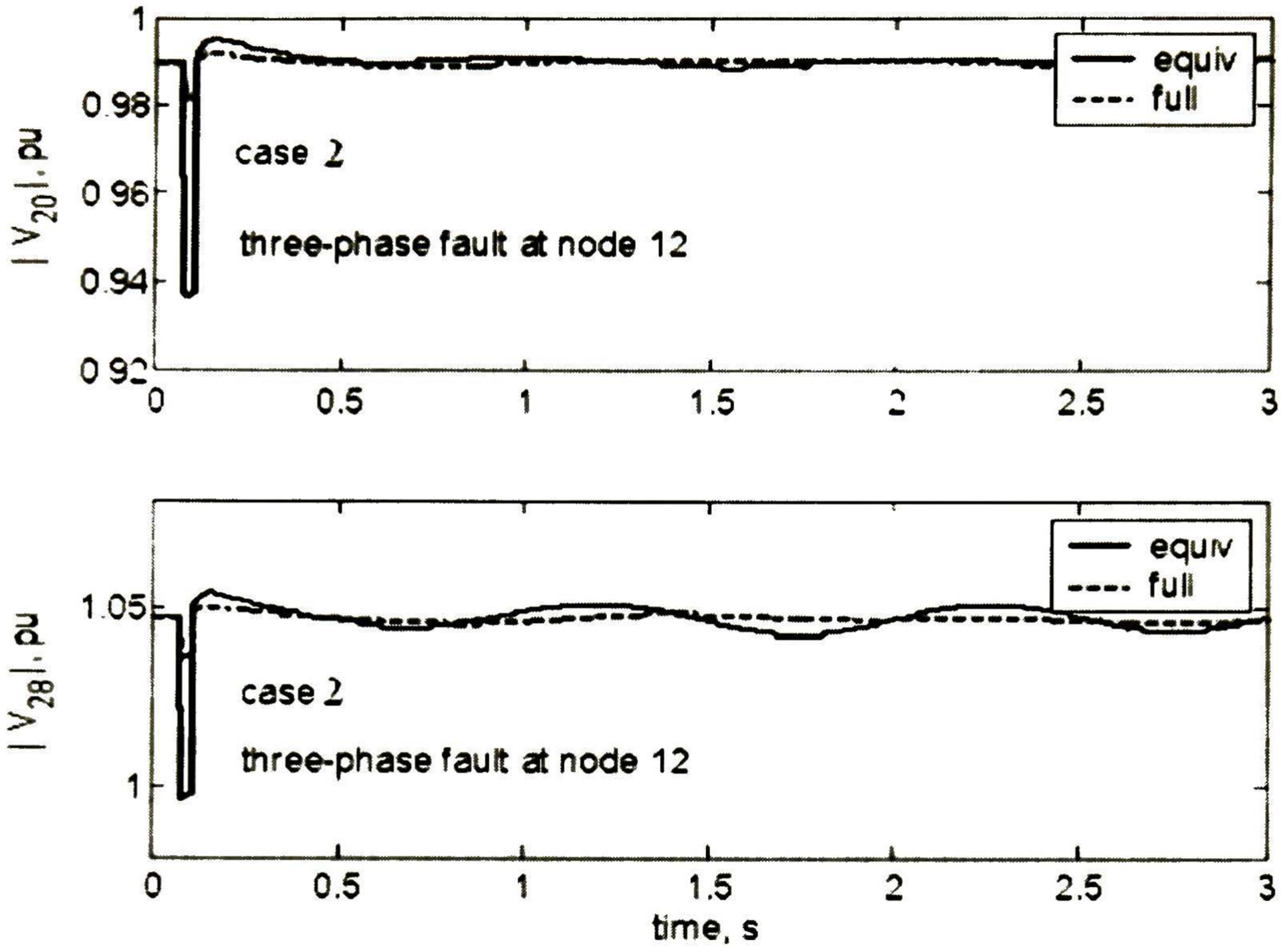



Fig. 4.3 Fault at node 12. Voltage magnitudes $|V_{28}|$ and $|V_{20}|$. Case 2.

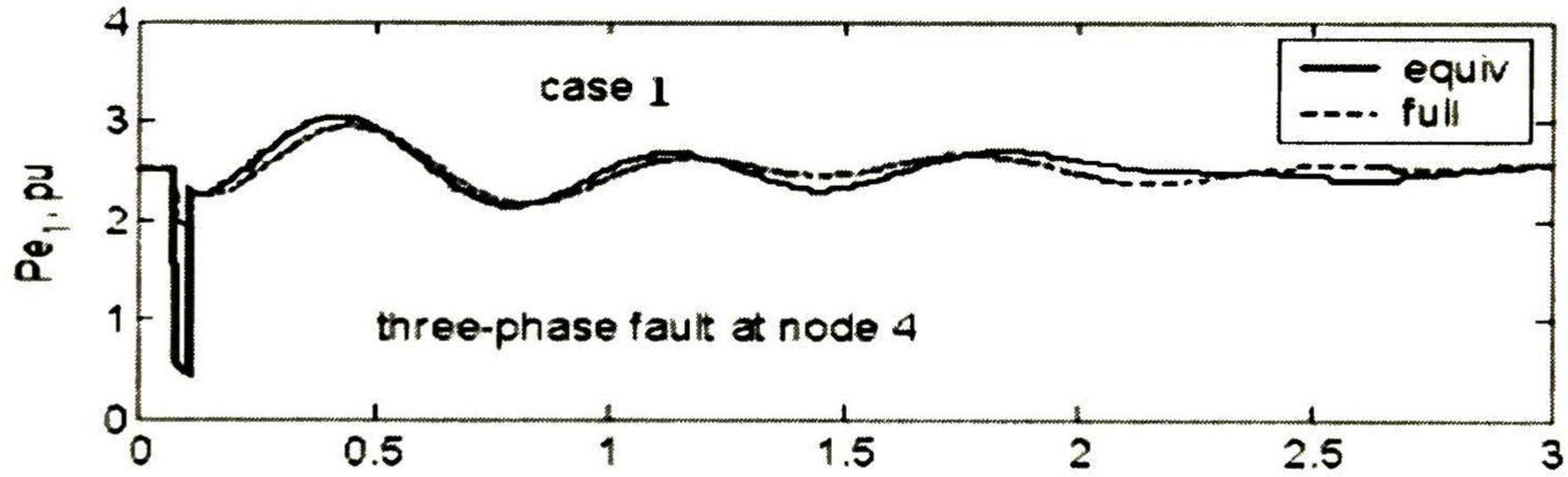



Fig. 4.4 Faults at node 4 and 9. Electrical power $P_{e1}$ and $P_{e9}$. Case 1.

Table 4.3 RMS errors (fault at node 19, *case 3*)

| $\delta$ (degrees) | $\Omega$ (rad/seg) | Pe (pu) |
|---|---|---|
| 2. 1742 | 0. 2021 | 0. 3849 |
| 2. 3979 | 0. 2199 | 0. 2738 |
| 2. 2887 | 0. 1868 | 0. 2371 |
| 3. 0174 | 0. 2394 | 0. 5273 |
| 2. 9488 | 0. 2351 | 0. 3903 |
| 3. 1237 | 0. 2438 | 0. 3464 |
| 2. 9928 | 0. 2277 | 0. 2590 |
| 2. 1260 | 0. 1818 | 0. 2876 |
| 4. 5027 | 0. 4043 | 0. 5421 |

From Table 4.3 can be deduced that generators 4, 6 and 9, Figs. 4.2 exhibit the larger angular position deviation, respect to the full system, for that fault. Despite the modal equivalent is derived based on linearized systems, the behaviour of the reduced network under non-linear simulations (transient stability) can be judge as appropriate.

### 4.2.6 Including stabilizer.

Besides static exciters, generators are equipped with power system stabilizer of the type

$$\frac{y(s)}{u(s)} = \frac{skT}{1+sT} \cdot \frac{1+sT_1}{1+sT_2} \cdot \frac{1+sT_3}{1+sT_4} \tag{4.12}$$

whose parameters are: $k = 0.1$, $T = 7.5$, $T1 = T3 = 0.045$, $T2 = T4 = 0.015$. In this case, problem (4.4) is solved by GA, with the estimated parameters for the fictitious generators showed in Table 4.4. Fig. 4.5 to 4.7 depict a sample of angles, velocities, electrical power and voltages, comparing the behaviour of the full and reduced system after a three-phase fault, under different operating conditions and fault locations. Table 4.5 shows an example of the RMS errors encountered in this case. Through the analysis of figures and RMS errors, can be deduced that GA are able to obtain better results that those showed in section 4.2.5. That is, the behaviour of the equivalent signals is closer to that of the full ones; in Fig. 4.6 should be noted the ordinates scale.

Table 4.4 Estimated parameters for the equivalent generator

| Generator Eq. 1 | Generator Eq. 2 |
|---|---|
| $X_{d\_1} = 1.0098$ | $X_{d\_2} = 0.7515$ |
| $X'_{d\_1} = 0.1500$ | $X'_{d\_2} = 0.0001$ |
| $T'_{d0\_1} = 6.2743$ | $T'_{d0\_2} = 5.4690$ |
| $X_{q\_1} = 0.4522$ | $X_{q\_2} = 1.4384$ |
| $X'_{q\_1} = 0.0388$ | $X'_{q\_2} = 0.1500$ |
| $T'_{q0\_1} = 12.985$ | $T'_{q0\_2} = 36.078$ |
| $H_1 = 1412.6$ | $H_2 = 390.53$ |
| $D_1 = 5.4462$ | $D_2 = 0.4068$ |
| $K_{a\_1} = 254.47$ | $K_{a\_2} = 513.38$ |
| $T_{a\_1} = 2.4782$ | $T_{a\_2} = 1.8333$ |

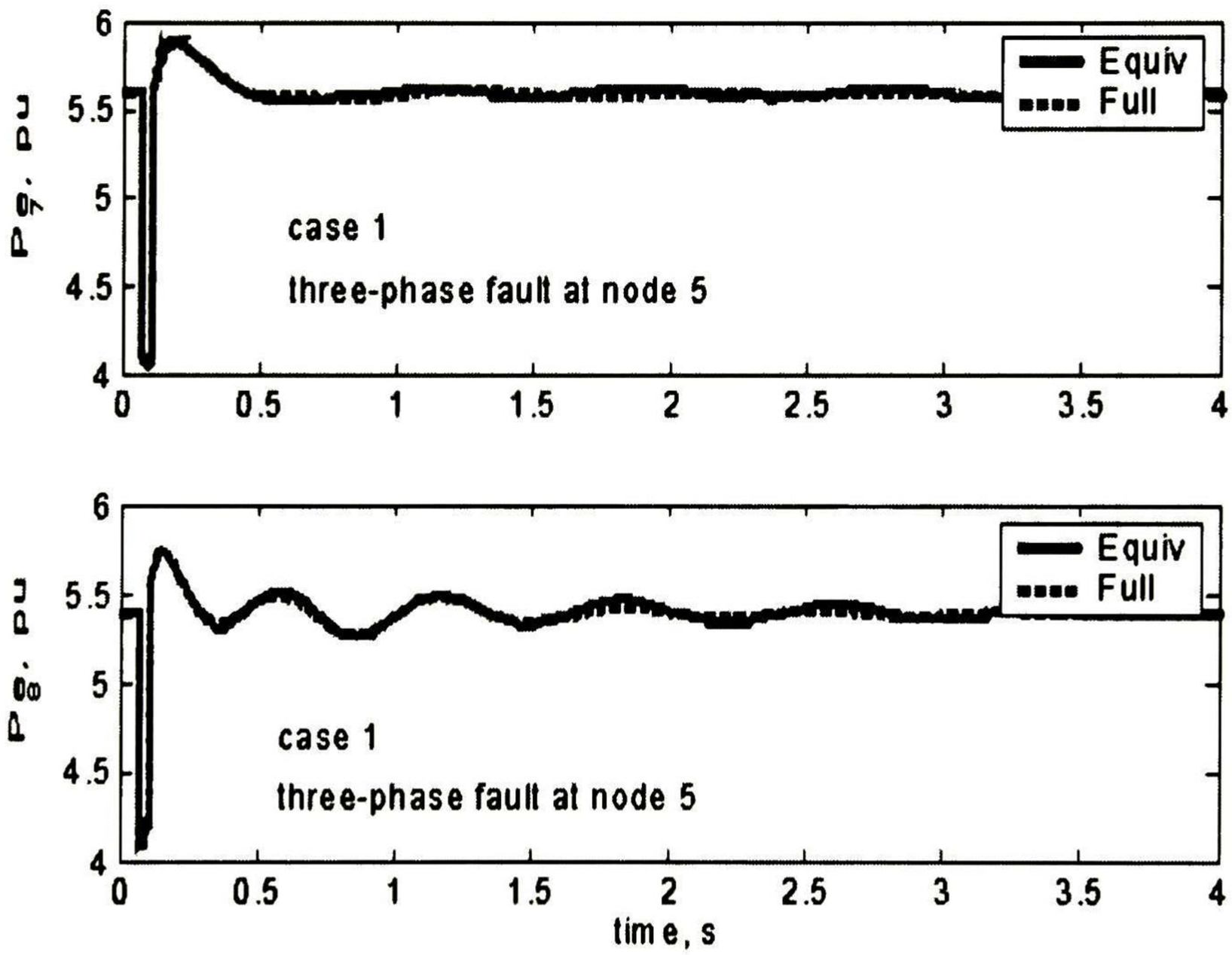
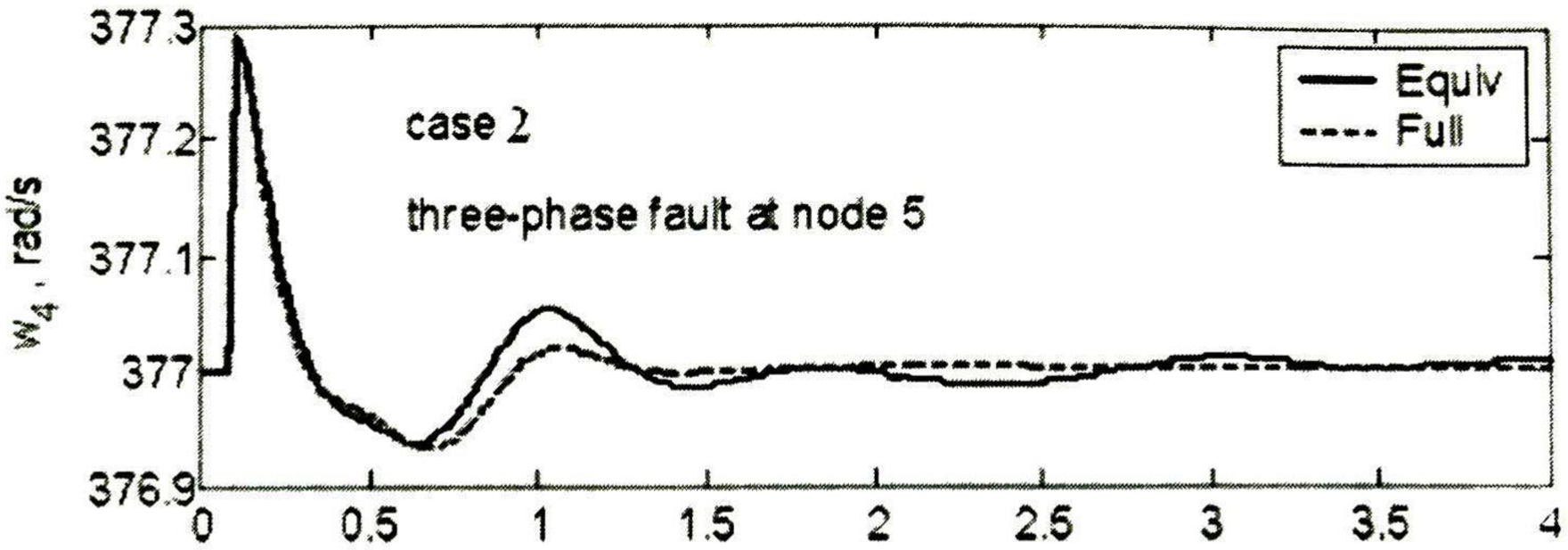



Fig. 4.5 Electrical power $P_{e7}$ and $P_{e8}$. Case 1. Fault at node 5.



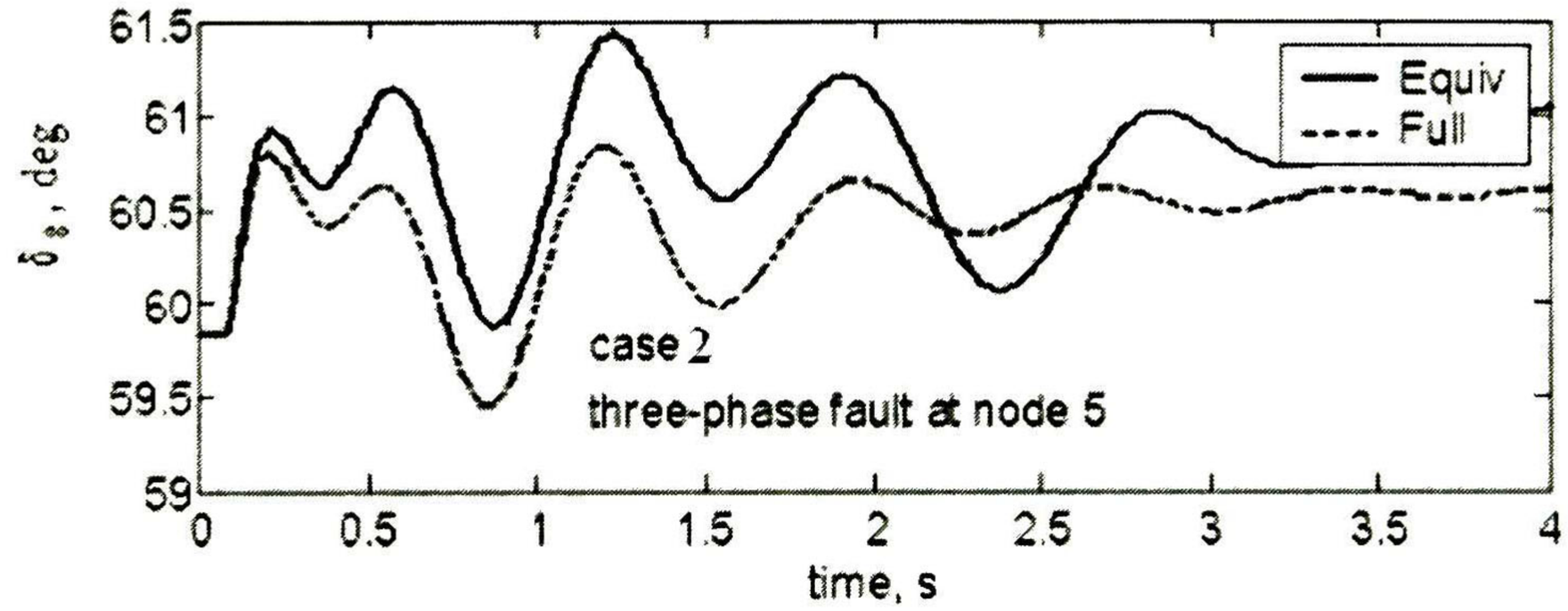
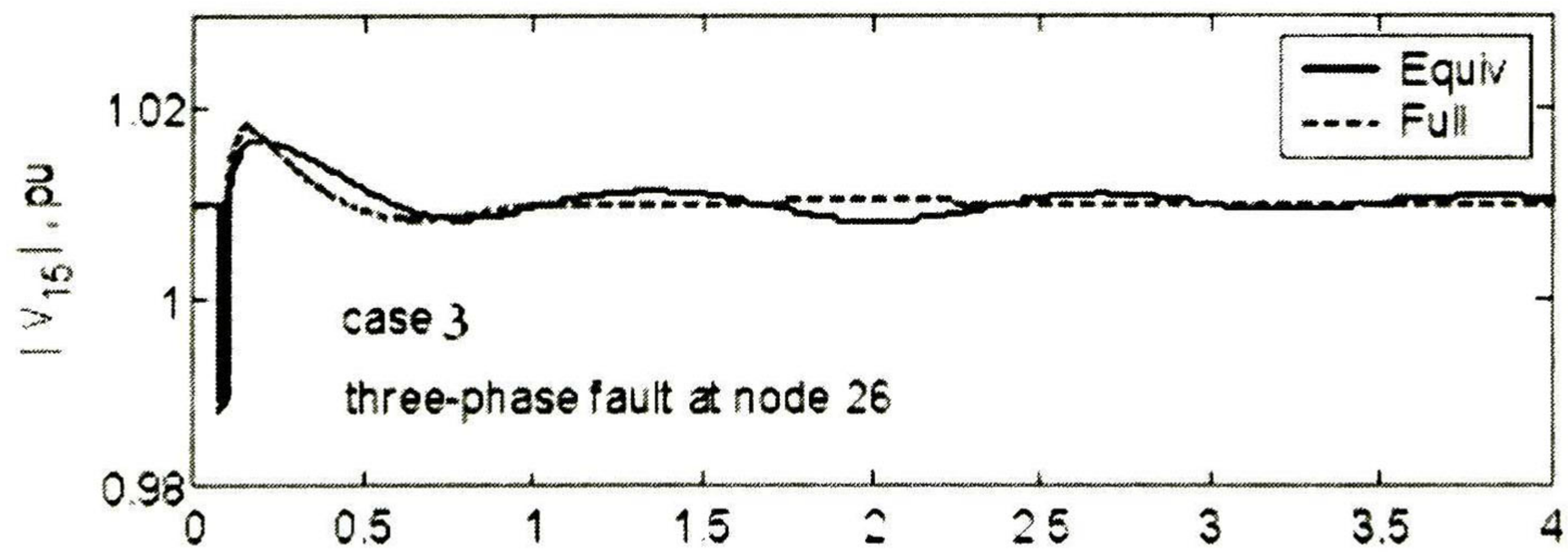




Fig. 4.6 Angular velocity $\omega_4$ and angular position $\delta_8$. Case 2. Fault at node 5.



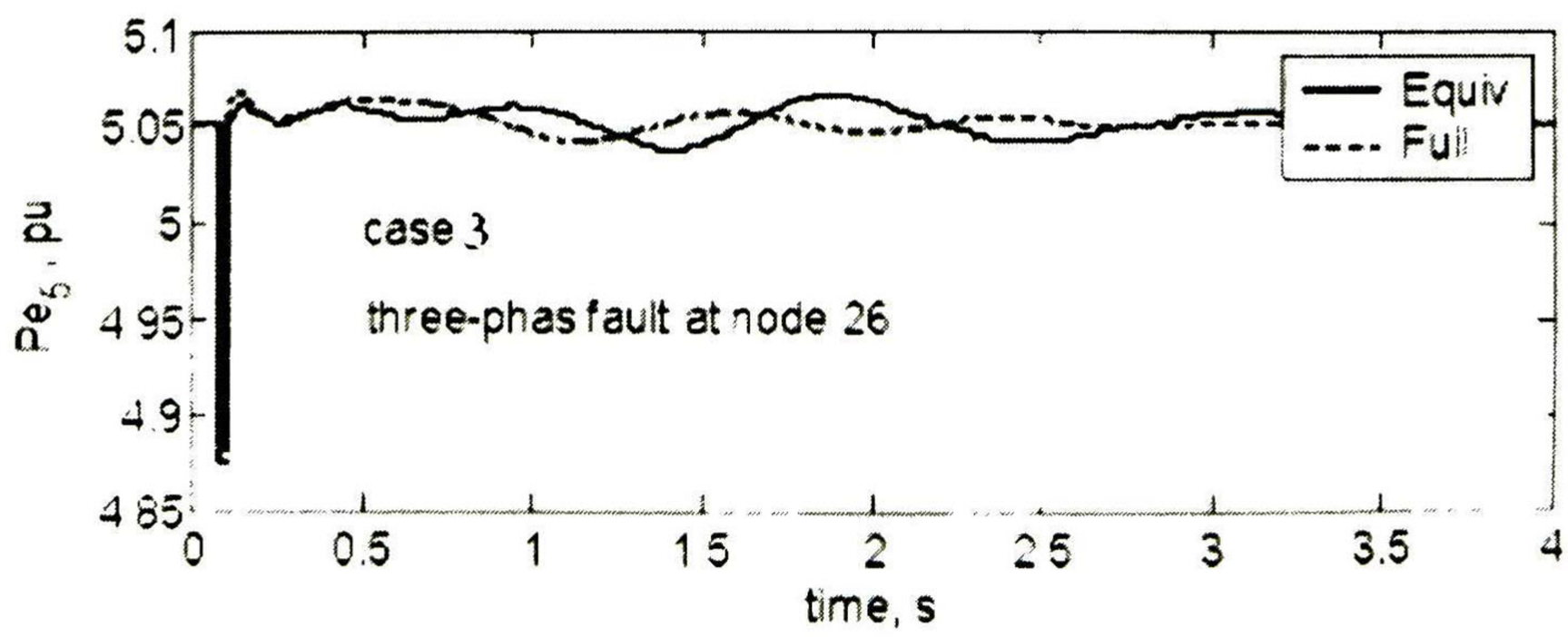




Fig. 4.7 Electrical power $P_{e5}$ and voltage $|V_{15}|$. Case 3. Fault at node 26.

Table 4.5 RMS errors (fault at node 5, *case 3*)

| $\delta$ (degrees) | $\Omega$ (rad/seg) | Pe (pu) |
|---|---|---|
| 0.5471 | 0.0424 | 0.0707 |
| 0.5706 | 0.0328 | 0.0476 |
| 0.5465 | 0.0253 | 0.0432 |
| 0.5018 | 0.0148 | 0.0144 |
| 0.5023 | 0.0154 | 0.0123 |
| 0.5122 | 0.0177 | 0.0211 |
| 0.5078 | 0.0161 | 0.0146 |
| 0.4731 | 0.0245 | 0.0237 |
| 1.1487 | 0.1031 | 0.1198 |

From Table 4.5 can be deduced that generator 9, display the larger angular position deviation, respect to the full system, for that fault.

## 4.3 Looking for a Neural Equivalent.

In this section, the possibility of predicting the terminal voltage through a neural net is explored. With that purpose, a single machine infinite bus (SMIB) is employed, where the model of the synchronous machine is third order (synchronous machine models are described at section 2.3) and it is equipped with a static system excitation, including a flexible alternate current transmission systems (FACTS) device which has been selected to be a unified power flow controller (UPFC) embedded into the transmission line [26], Fig. 4.8.

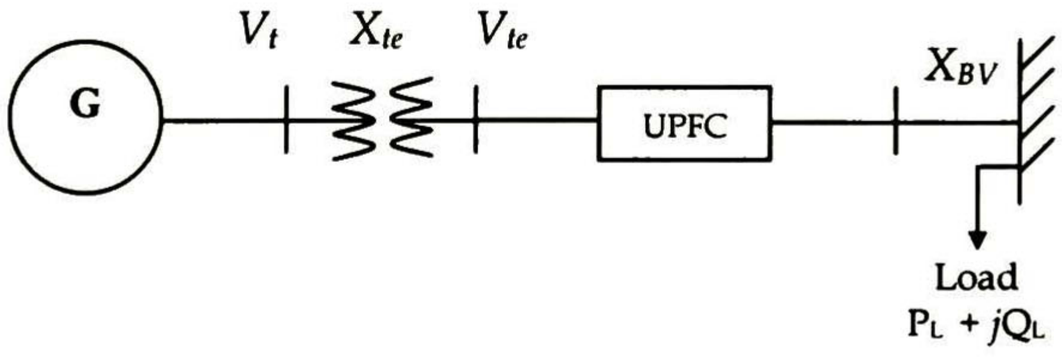



Fig. 4.8 Single machine infinite-bus.

In the following, the parameters and their values (expressed in per unit, except frequency) that are established for this system are described.

Active power load. $P_L = 3$;

Reactive power load. $Q_L = 1.25$;

Transformer Reactance. $X_{tE} = 0.015$;

Transmission line Reactance. $X_{Bv} = 0.02$;

Damping factor. $D = 0$;

Inertia constant. $H = 3.5$;

Machine synchronous reactance in d-axis. $X_d = 0.0525$;
Machine synchronous reactance in q-axis. $X_q = 0.05$;
Machine transient reactance in d-axis. $X'_d = 0.010$;
Machine transient reactance in q-axis. $X'_q = 0.05$;
UPFC series reactance $X_E = 0.005$;
UPFC shunt reactance from $X_B = 0.005$;
Time transient constant $T'_{d0} = 5.0$;
Excitation system parameters: Ka = 25; Ta = 0.05;
Frequency. 60 Hertz

Initial conditions are taken from a steady-state power flow study.
Vt = 1.0456 + j 9.5857 e-2
Vte = 1.0337 + j 5.1728 e-2
VE = 1.0360 + j 5.2019 e-2
VB = 2.4565 e-3 – j 2.3272 e-2;
Vt is the terminal voltage.
Vte is the transformed terminal voltage.
VE is the voltage at the UPFC shunt source.
VB is the voltage at the UPFC series source.

With the purpose of training a NN able to reproduce the terminal voltage *Vt*, the system is perturbed with 4 variations in the transmission line reactance. These disturbances are considered due to they result the most convenient for our purposes. Fig. 4.9 and Fig. 4.10 show the angular position and the rotor's angular velocity when these disturbances are applied.

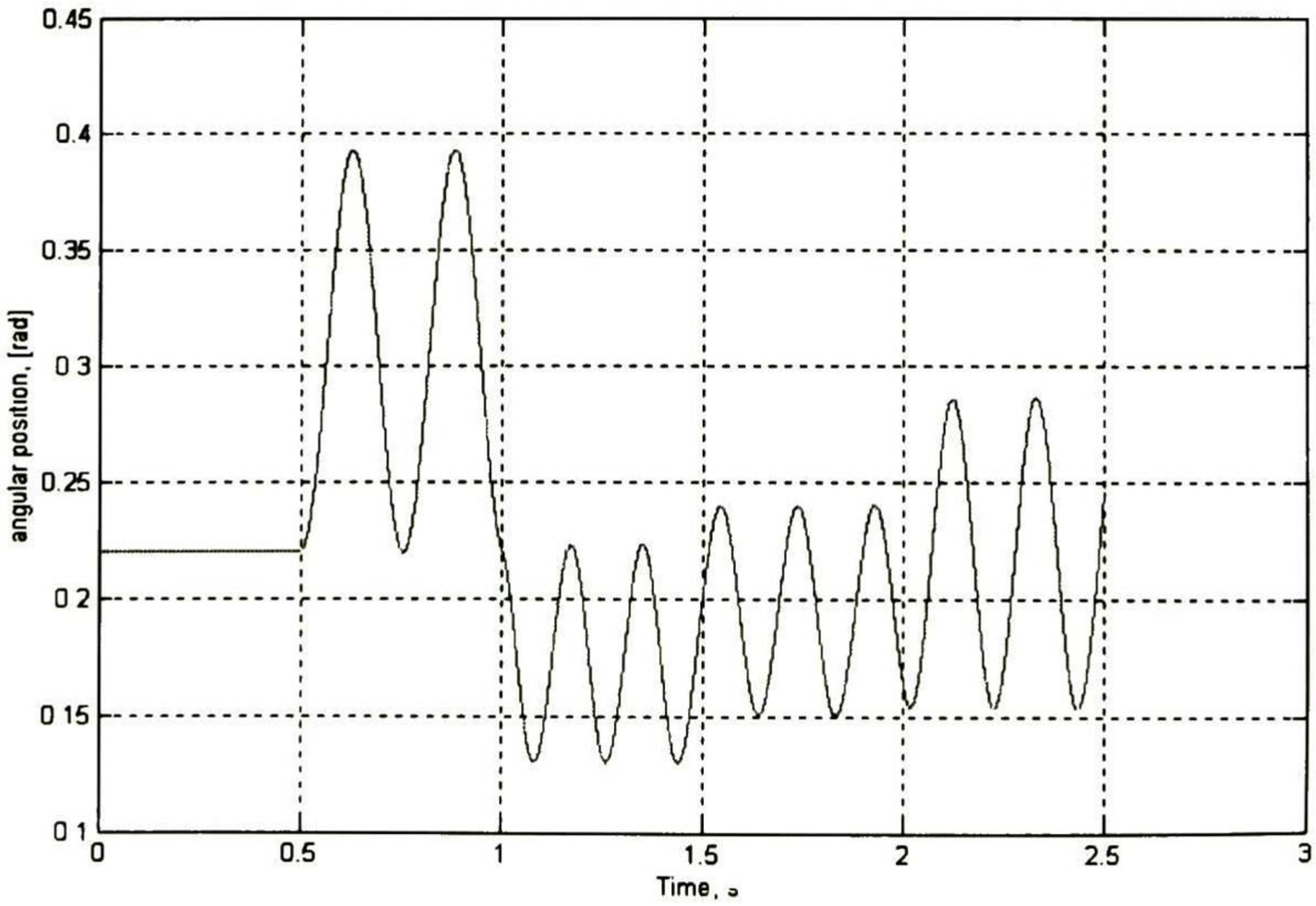



Fig. 4.9 Rotor machine's performance when a transmission line reactance modification is simulated.

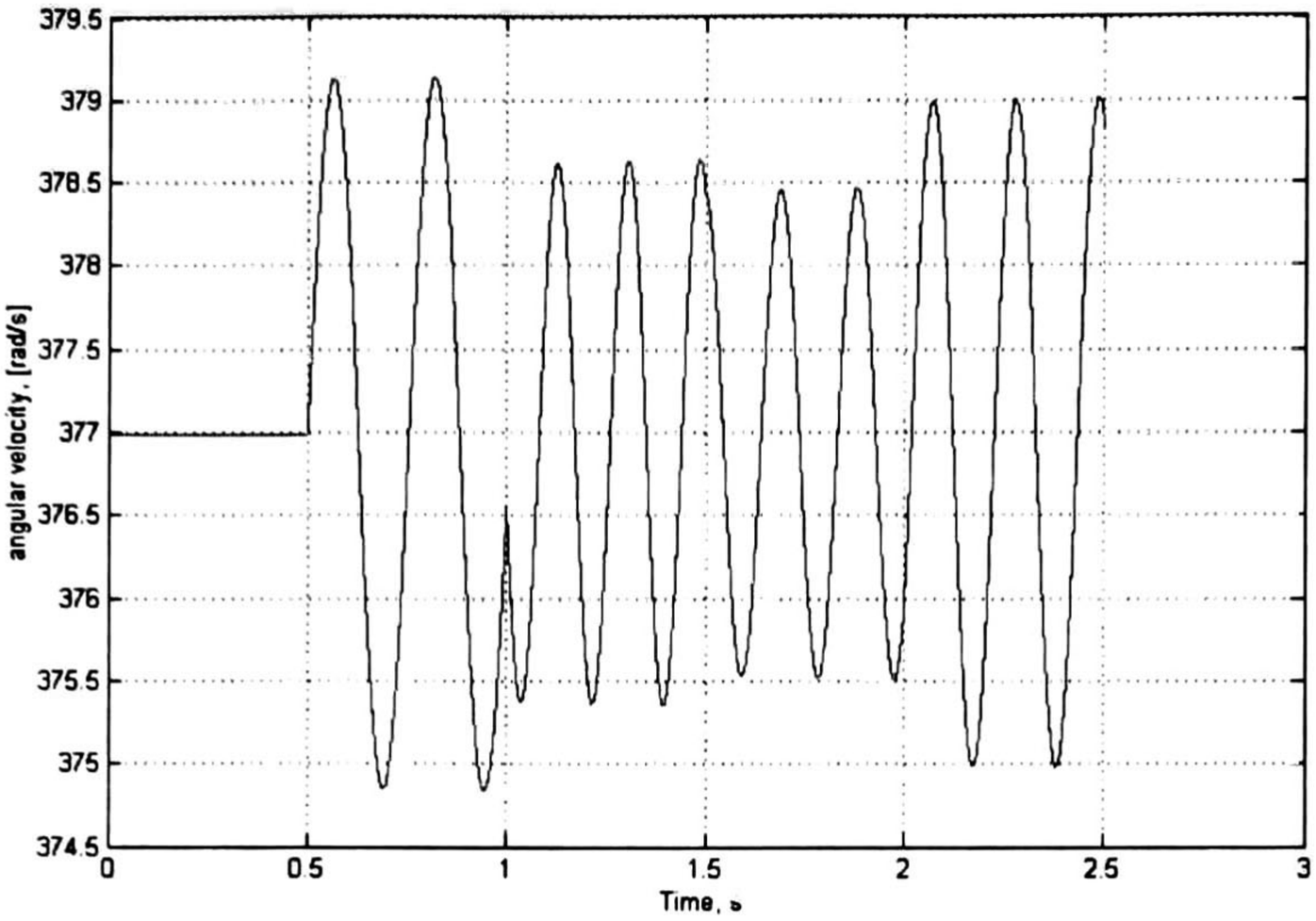



Fig. 4.10 Angular velocity performance when a transmission line reactance disturbance is applied.

The first disturbance is applied at 0.5 seconds, the second one at 1.0 second and the others at 1.5 and 2.0 seconds, respectively. The total simulation time is 2.5 seconds. From this simulation, some values are observed, such as the electric torque, $T_e$; the terminal voltage $Vt_r + jVt_i$; and the rotor angle deviation, $\delta$. These values are capturing and employing to train an ANN that possesses the following features. It has two input layers, one hidden layer with two neurons that are activated with a sigmoid function (tangent-hyperbolic function), and two output layers that have a linear activation function. The input values are the electric torque $T_e$, and the rotor angle deviation $\delta$. The output values are the terminal voltage, both real and imaginary values. This is done by reason of the ANN that is used in not able to give high accuracy results employing the phase voltage as a complex number. The ANN is trained with the Levenberg-Marquardt method, the weights from input to hidden layer and hidden to output are initialised randomly and it is trained for 1000 epochs. Intentionally, all data are trained; that is, the training is done for the complete simulation time. Afterwards, and having all data trained, a prediction is done. This prediction consists in foreseeing the terminal voltage *Vt*, when another disturbance perturbs the system. The infinite bus voltage drop, the transmission line reactance and the load change. These results are exemplified in Fig. 4.11, Fig. 4.12 and Fig. 4.13, respectively. For these disturbances, simulation time is for two second, the faults are applied at 0.5 second from starting time and they are cleared at 0.5 seconds after applying the fault. Figs. 4.14a, 4.14b and 4.14c exhibit the ability of ANN for solving the problem; that is, predict the terminal voltage. The prediction is performed as the same way as it is done in example 6, section 3.4 from Chapter 3. It is important to remark that for this prediction one of the past outputs and one of the past inputs are used for determining the prediction so as the time delay is zero.

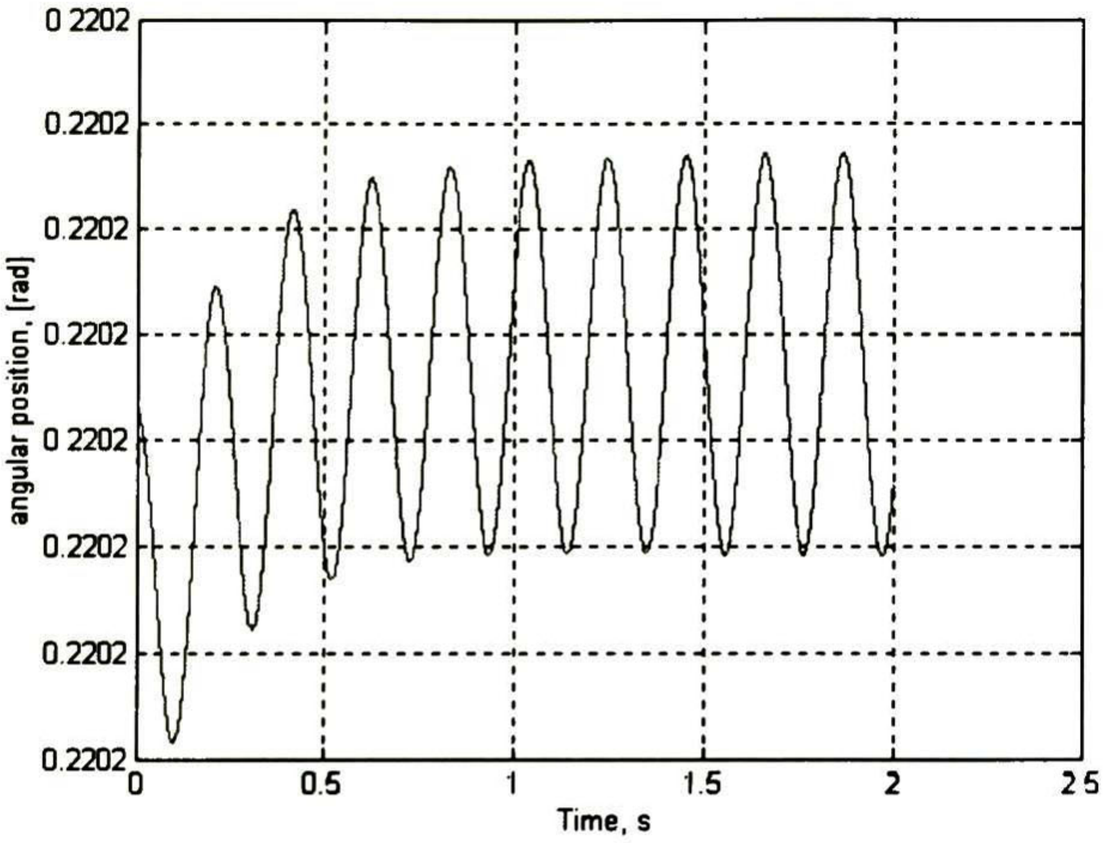



Fig. 4.11a Rotor machine's performance when a load variation disturbance is applied.

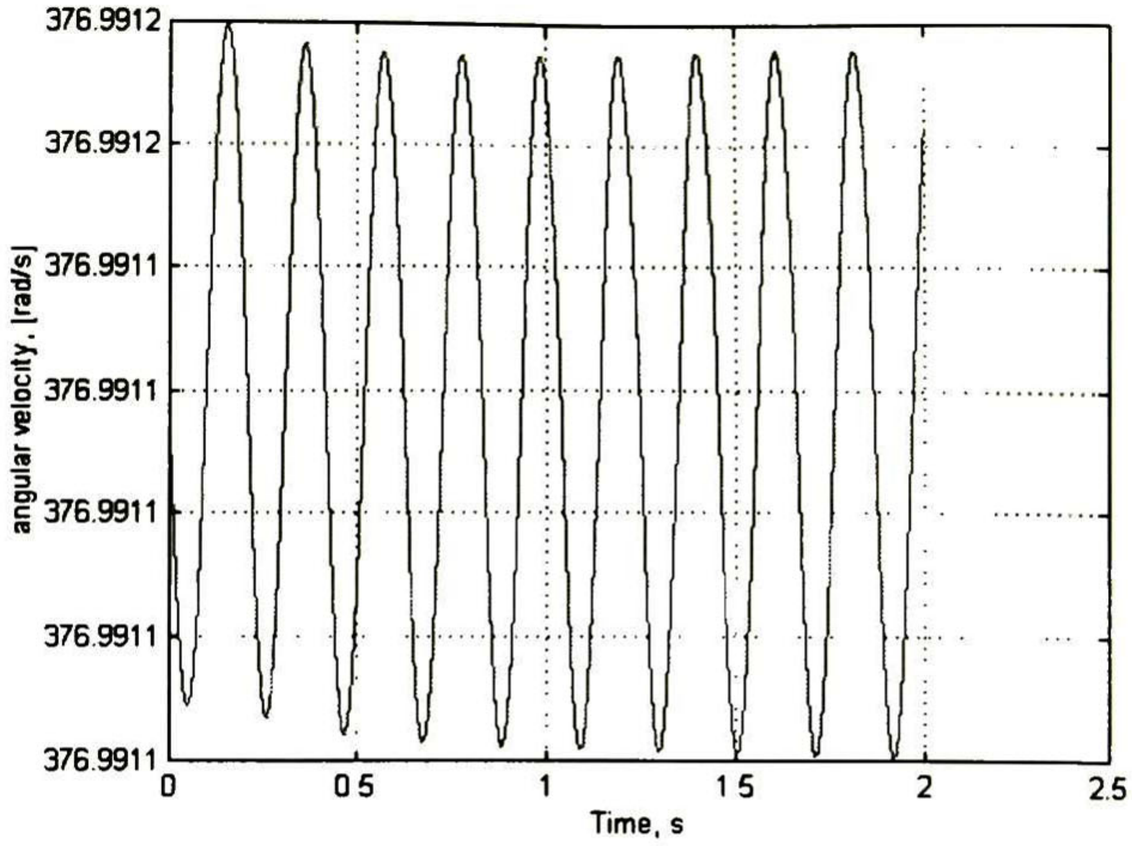



Fig. 4.11b Angular velocity performance when a load variation is applied.

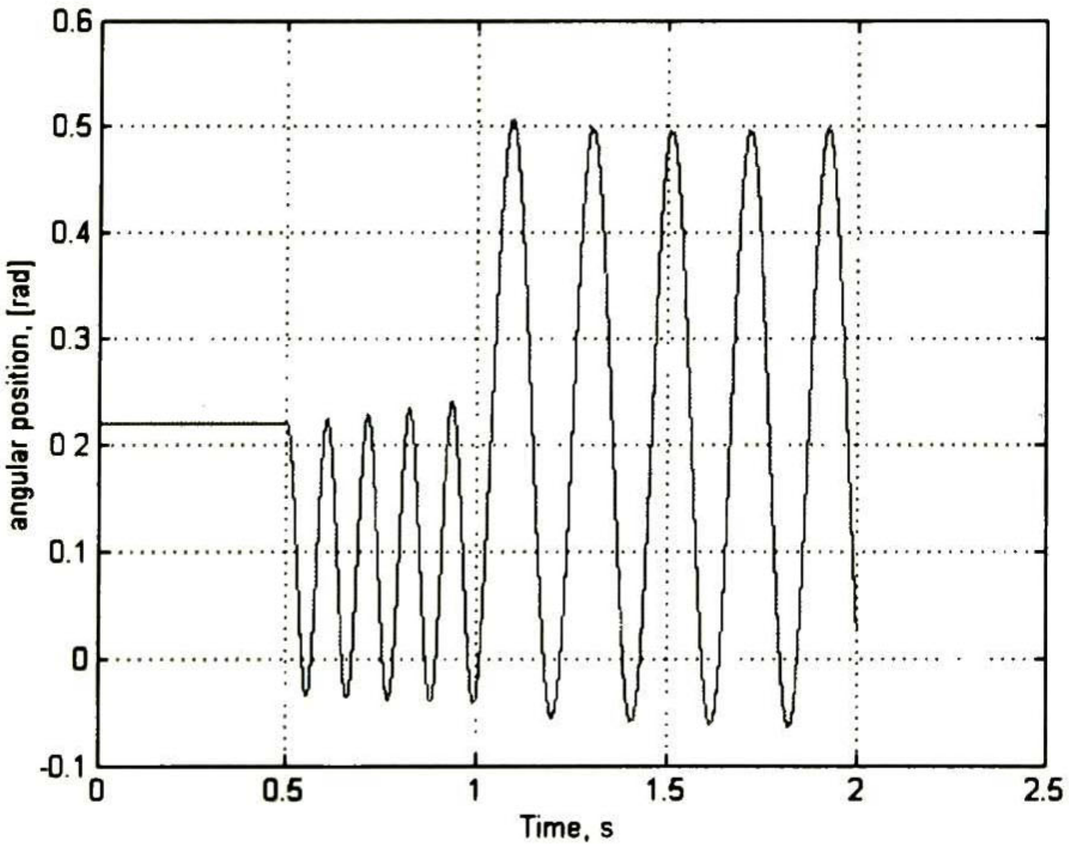



Fig. 4.12a Rotor machine's performance when a transmission line reactance change is applied.

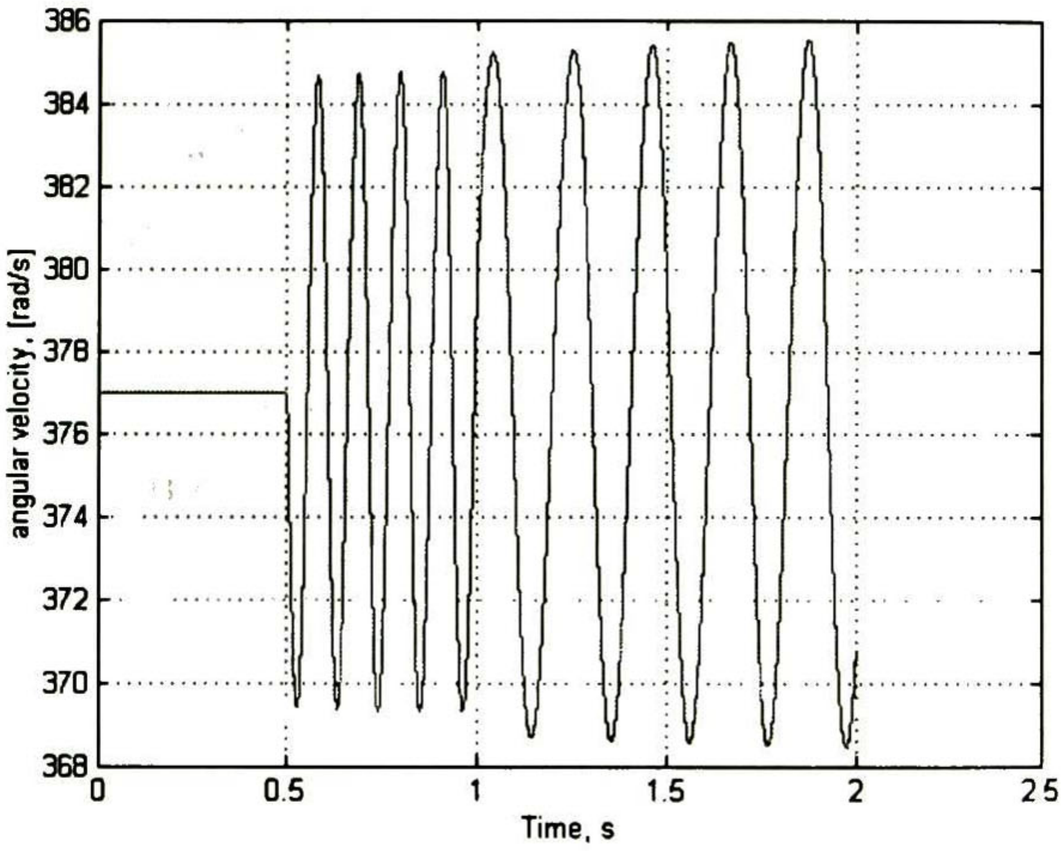



Fig. 4.12b Angular velocity performance when a transmission line reactance change is applied.

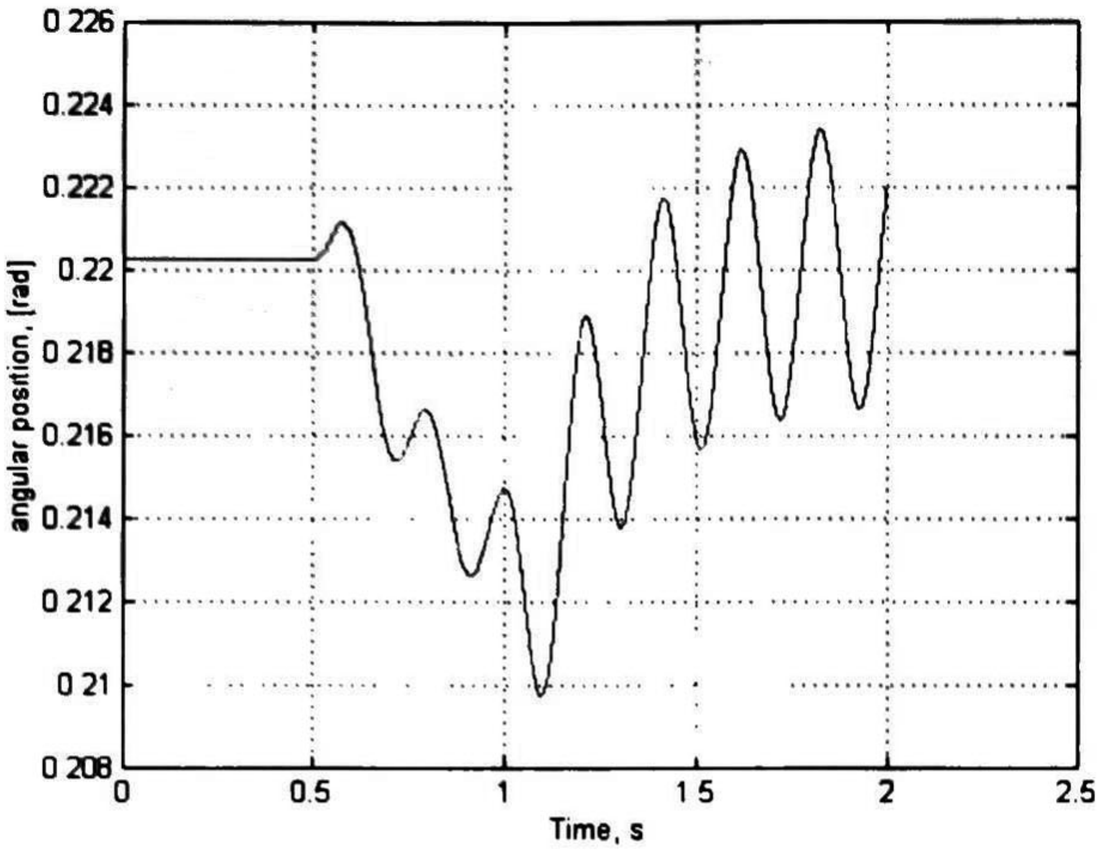



Fig. 4.13a Rotor machine's performance when a load variation is applied.

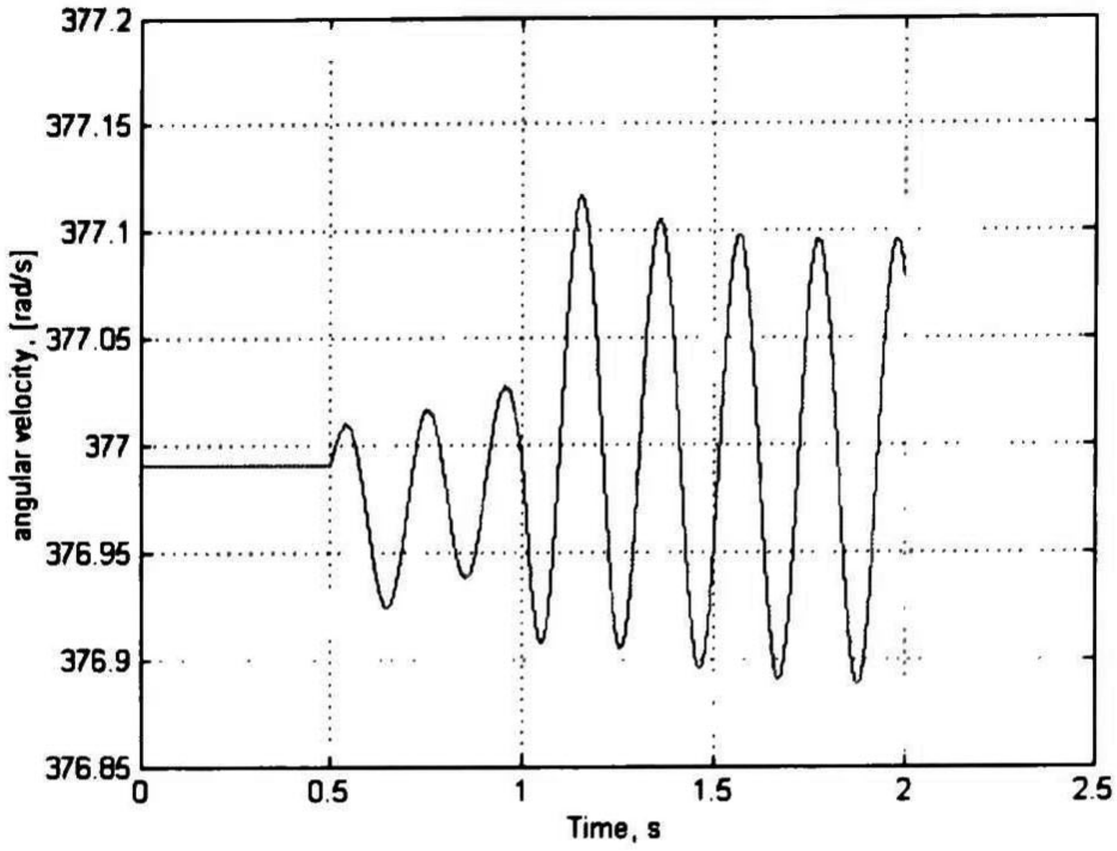



Fig. 4.13b Angular velocity performance when a load variation is applied.

Figs. 14a, 14b and 14c are the relationship between the actual outputs that are taken from the transient stability programme and the outputs that are obtained from the trained NN. The solid lines are the transient stability programme outputs and the dash lines are the predicted ones.

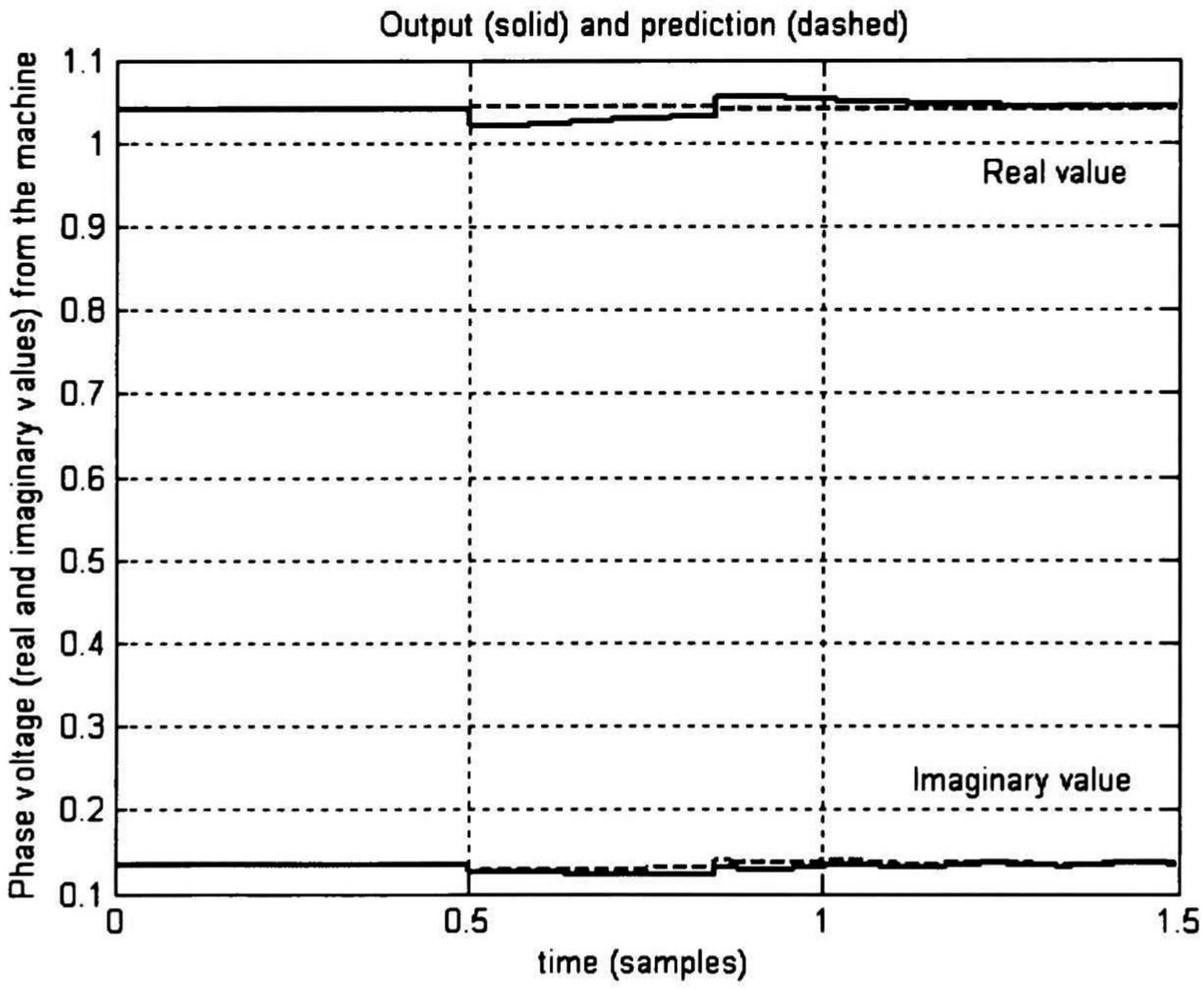



Fig. 4.14a Terminal voltage.

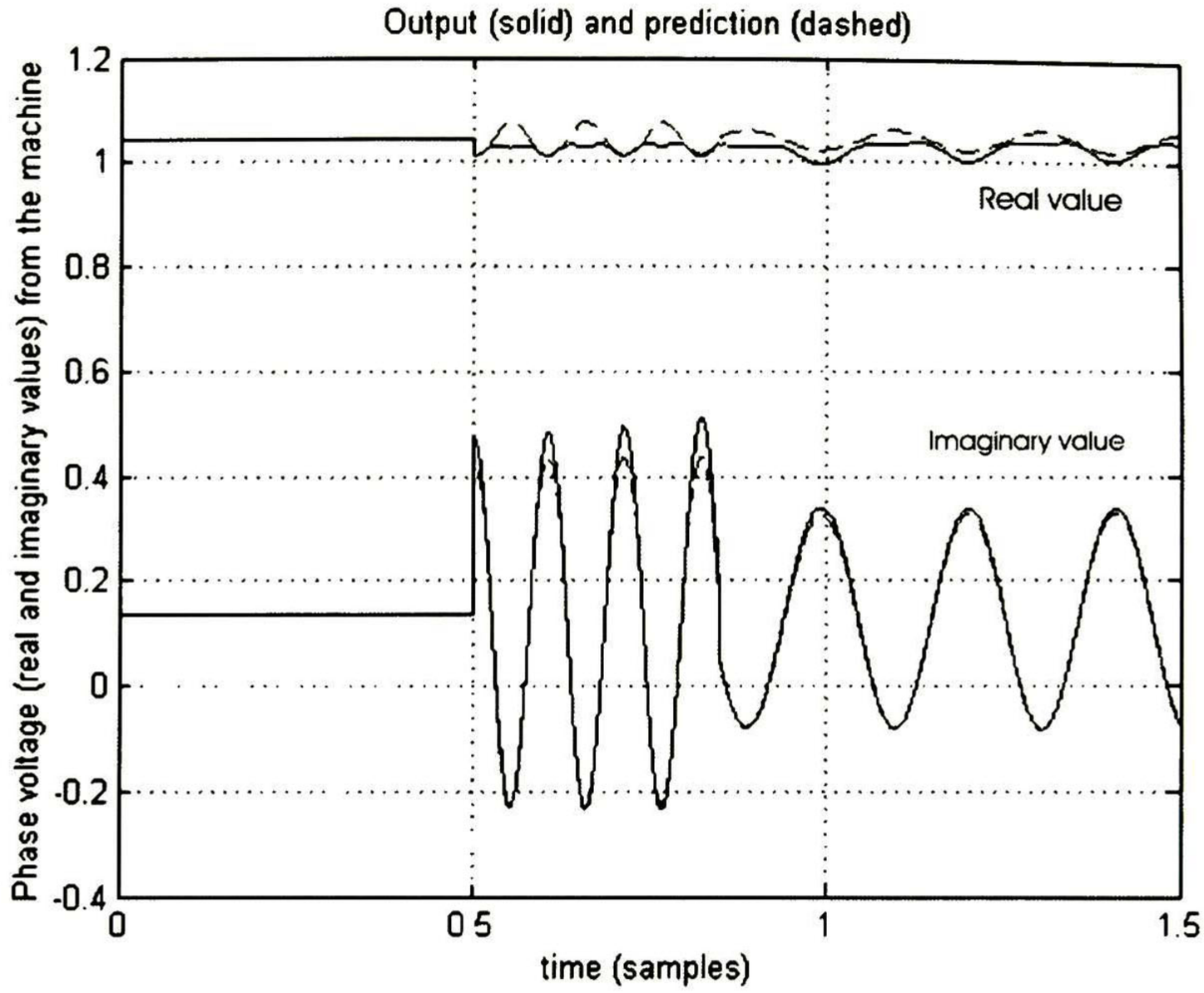



Fig. 4.14b Terminal voltage under transmission line reactance change.

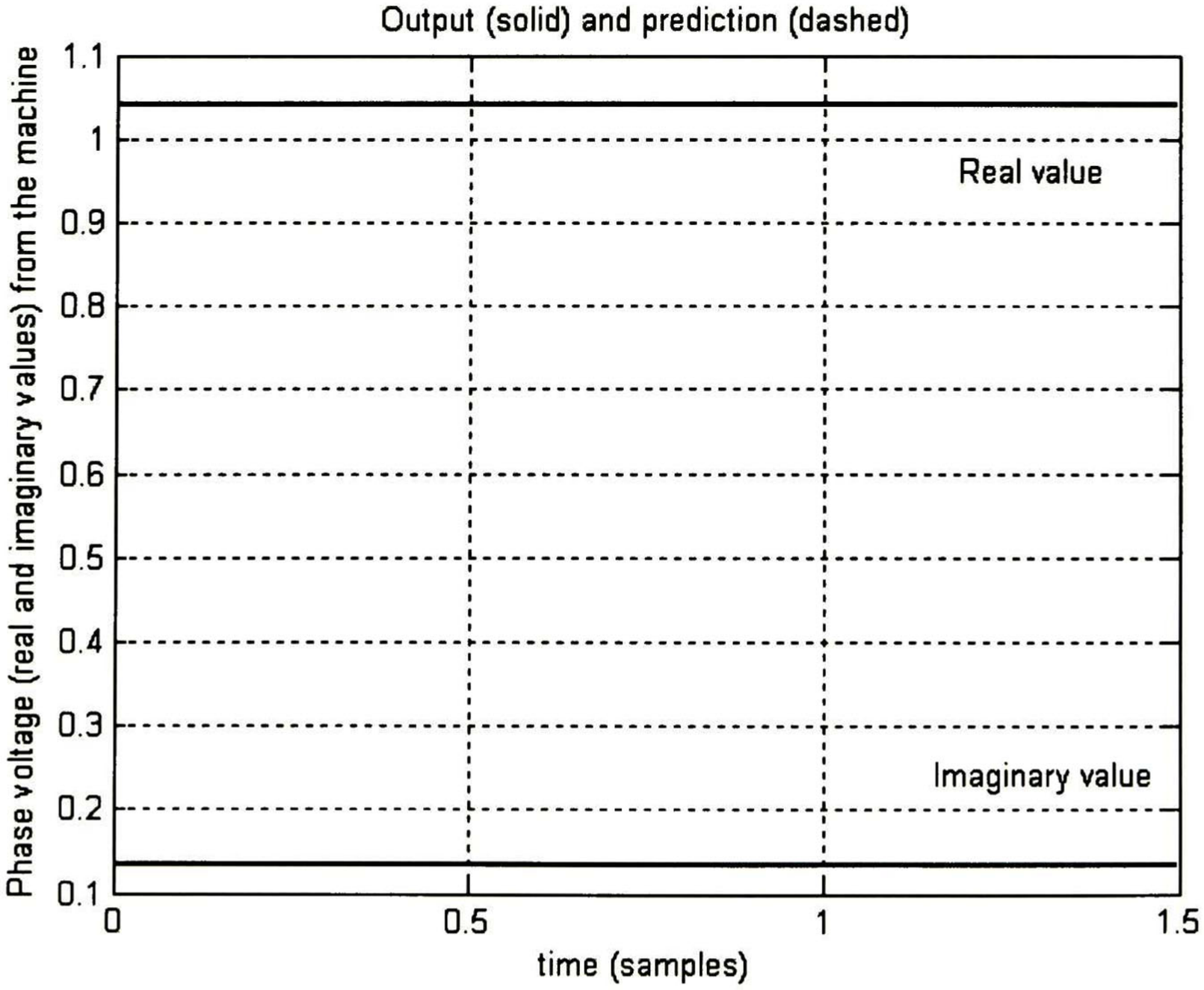



Fig. 4.14c Terminal voltage under load change.

Even when the trouble to be solved is a high degree difficult one, due to the disturbance is different from the one that is applied for predicting, the results demonstrated the capability and the high accuracy results of this artificial neural network for forecasting the terminal voltage.

### 4.3.1 3 Machines Power System.

Next example is applied to a multi-machine power system, formed by 3 machines and 9 buses depicted in Fig. 4.15. The principal target in this example is to forecast the Bus voltage at node 4 using an artificial neural network. This one is trained in order to predict the complex voltage when the power system is perturbed with different disturbances from the ones used to train the neural network.

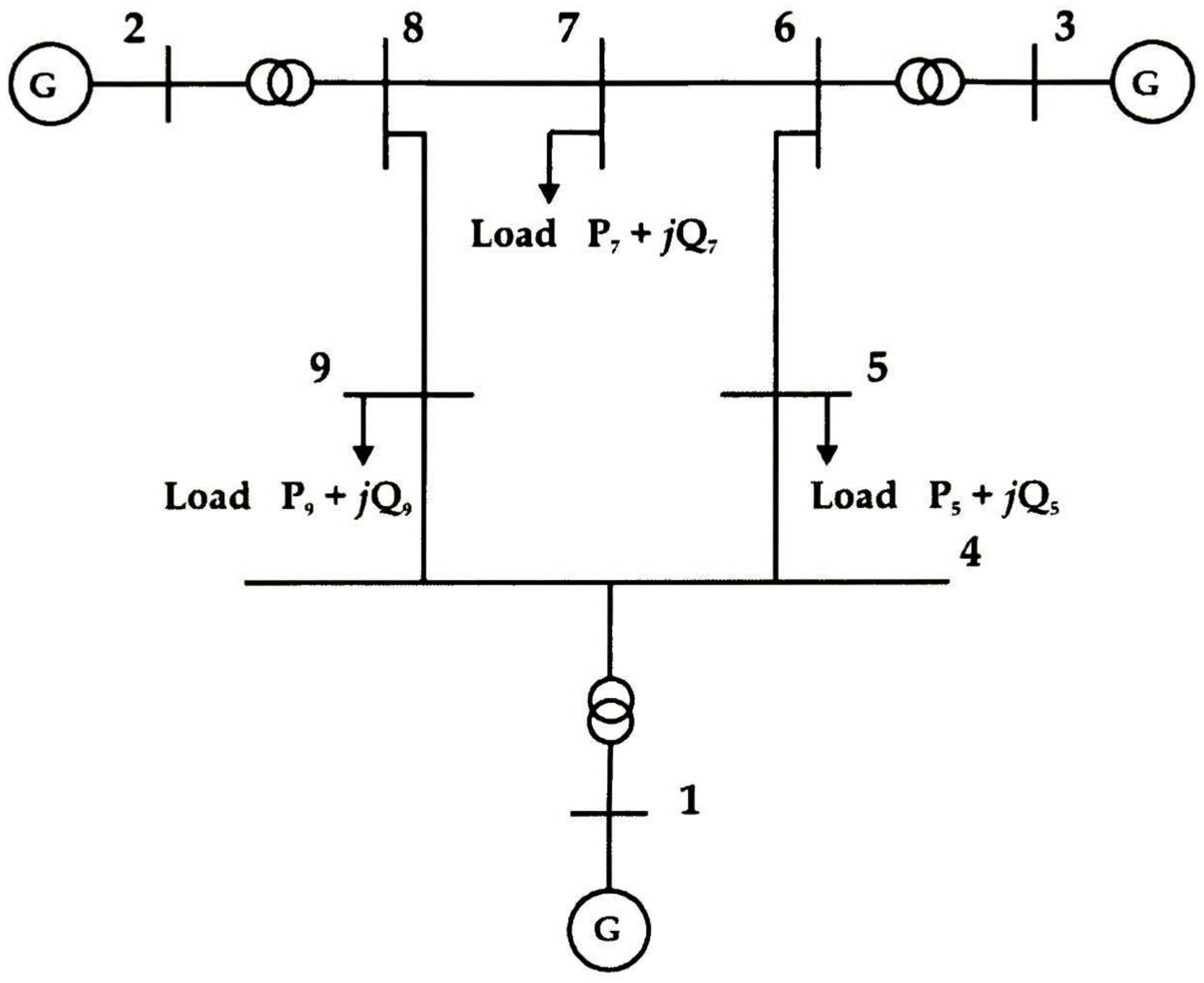



Fig. 4.15 3 Machines 9 buses power system.

The power system shown in Fig. 4.15 has the following description. Generators are represented by a fourth order model with a static excitation system, Eq. (4.2). Table 4.6 shows the generator's parameters used for this multi-machine system.

Table 4.6 Synchronous machine's parameters.

| Machine No. | $X_d$ | $X'_d$ | $T'_{d0}$ | $X_q$ | $X'_q$ | $T'_{q0}$ | $H$ |
|---|---|---|---|---|---|---|---|
| 1 | 0.1460 | 0.0608 | 8.96 | 0.0969 | 0.0569 | 0.100 | 23.64 |
| 2 | 0.8958 | 0.1198 | 6.00 | 0.8645 | 0.0969 | 0.535 | 6.4 |
| 3 | 1.3125 | 0.1813 | 5.89 | 1.2578 | 0.1500 | 0.600 | 3.01 |

At buses 5, 7 and 9 there are connected loads with the following values [27].

Table 4.7 Active and reactive power loads.

| | LOAD (pu) | |
|---|---|---|
| | **Active power P** | **Reactive power Q** |
| Bus 5 | **0.90** | ***j*0.30** |
| Bus 7 | **1.00** | ***j*0.35** |
| Bus 9 | **1.25** | ***j*0.50** |

The transmission lines parameters used for this multi-machine system are:

Table 4.8 Transmission lines parameters.

| TRANSMISSION LINES PARAMETERS | | | | | |
|---|---|---|---|---|---|
| From Bus No | To bus No | Resistance (pu) | Reactance (pu) | Line charging (pu) | Tap ratio |
| 1 | 4 | 0.0 | 0.0576 | 0.0 | 1.0 |
| 4 | 5 | 0.017 | 0.092 | 0.158 | 1.0 |
| 5 | 6 | 0.039 | 0.17 | 0.358 | 1.0 |
| 3 | 6 | 0.0 | 0.0586 | 0.0 | 1.0 |
| 6 | 7 | 0.0119 | 0.1008 | 0.209 | 1.0 |
| 7 | 8 | 0.0085 | 0.072 | 0.149 | 1.0 |
| 2 | 8 | 0.0 | 0.0625 | 0.0 | 1.0 |
| 8 | 9 | 0.032 | 0.161 | 0.306 | 1.0 |
| 4 | 9 | 0.01 | 0.085 | 0.176 | 1.0 |

4.3.1.1 Artificial neural network description.

The neural network used for training and prediction purposes has the next characteristics. It is a feedforward neural network that possesses 4 inputs, one hidden layer and 2 output layers. The hidden layer has 3 neurons that are activated with sigmoid functions (tangent-hyperbolic functions) and the output layers have a linear activation function. Before taking the decision to train the elected neural network, several arrays of neural networks were proved to solve the problem. That is, although other arrays present clear results when they are trained for a specific disturbances, they do not offer good answers when prediction is done for other fault different from the one that is used to train the neural network. Some other arrays have the same difficulty to do the prediction; thus, the neural network array that possesses the required and sufficient features to get optimum results is illustrated in Fig. 4.16.

The ANN is trained with the Levenberg-Marquardt method, the weights from input to hidden layer and hidden to output layer are initialised randomly and it is trained for 1000 epochs; even when it is trained for 1000 epochs, many times it is not necessary to wait until the algorithm reaches these epochs number because most of the times it converges promptly.

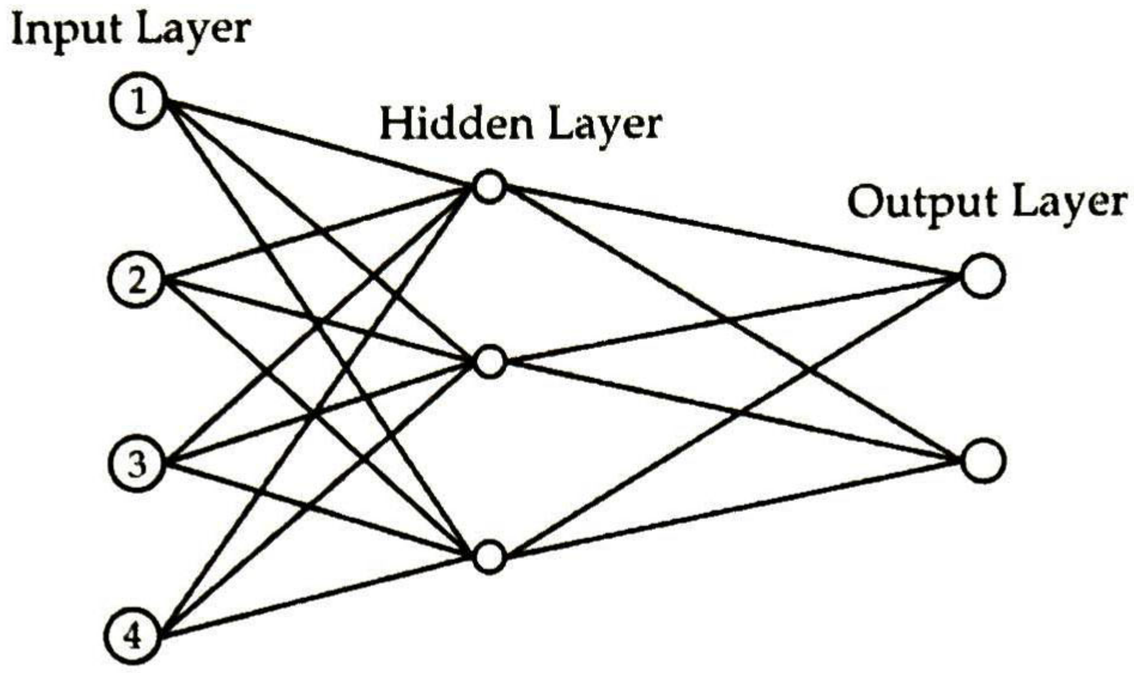



Fig. 4.16Artificial neural network used for training and prediction.

The prediction is done as in example 6, section 3.4, but it presents small differences. Now, the ARX model is used is a MIMO (multiple inputs - multiple outputs) ARX system. This model has the same distinctiveness attributes as the ARX model. As in the preceding example, it is important to mention that for doing the prediction only 1 past input and 1 past output are used; so, the prediction time delay is zero.

4.3.1.2 Training and predicting stage.

Initially, the system is perturbed with a three-phase fault at Bus 5. To simulate this fault, a large admittance is connected to ground. From 0 to 0.08s the system is in steady state, then at this time (0.08) the fault is applied and it is cleared at 0.13s. As it is noticed, the fault is for 3 cycles, going the system back to its original structure. The total time for this simulation is 4.0s. Figs. 4.17a, 4.17b and 4.17c show the behaviour of the three synchronous machines velocities, angular positions and electrical torque.

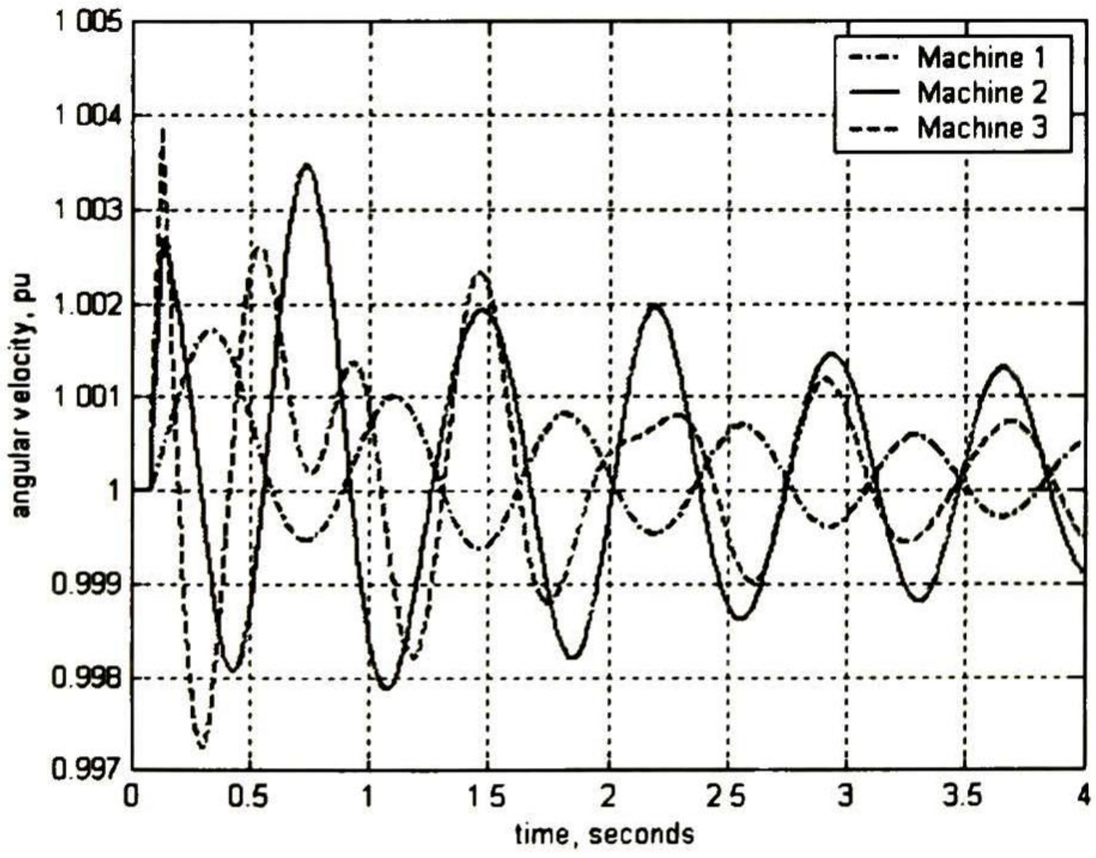



Fig. 4.17a Angular velocities under a three-phase fault at Bus 5.

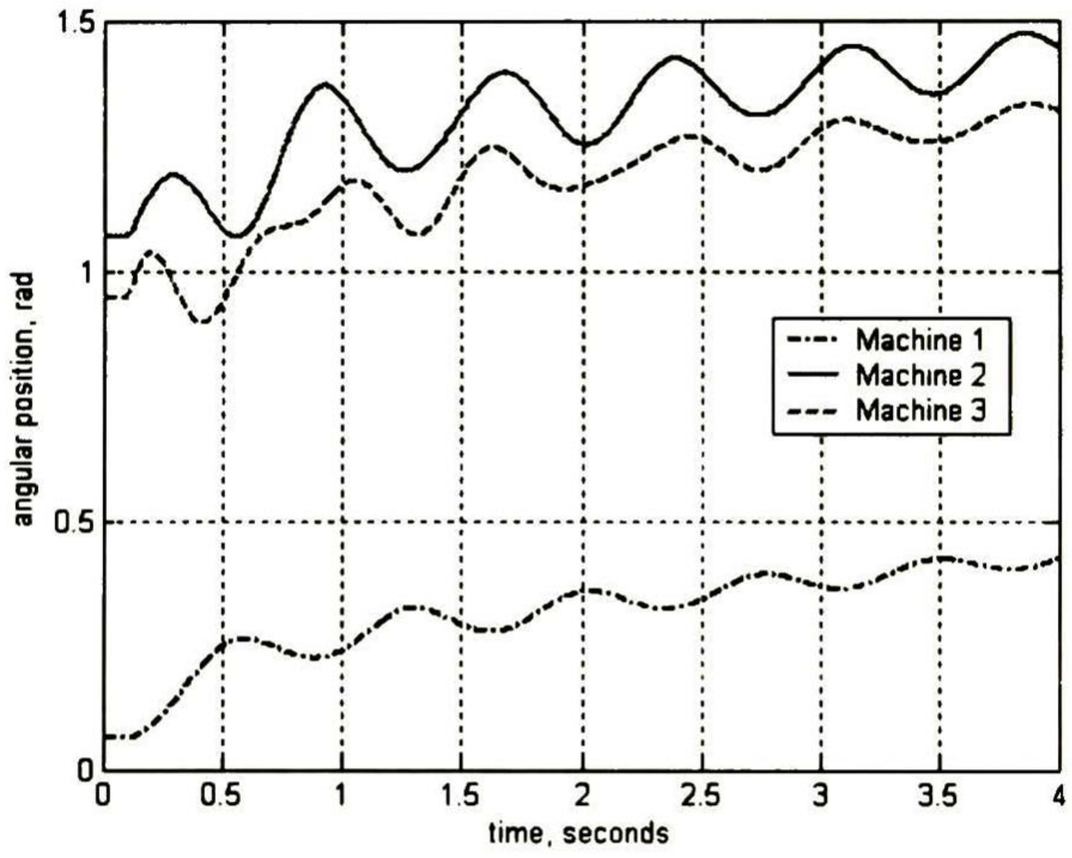



Fig. 4.17b Rotor's machine under a three-phase fault at Bus 5.

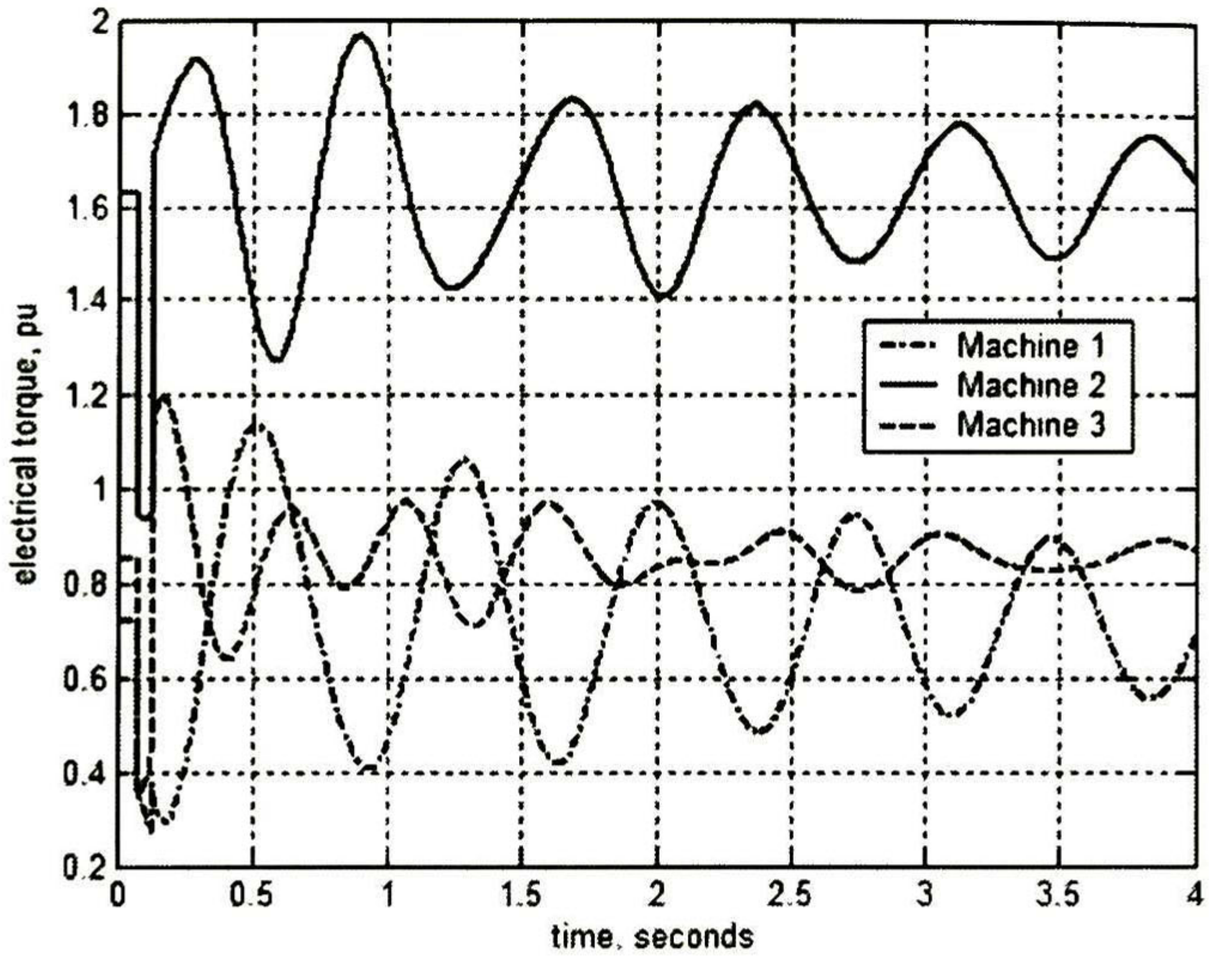



Fig. 4.17c Electrical torque under a three-phase fault at node 5.

The input values for the ANN are taken as the power flows from Bus 4 to 5 and from Bus 4 to 9 under the three-phase fault at node 5. Fig. 4.18 and 4.19 show the real and imaginary (active and reactive) power flows behaviour under this contingency. Formerly, many other transient stability simulations were done with the aim to obtain the best results able to train the neural network and this one offered the appropriate results that we were seeking.

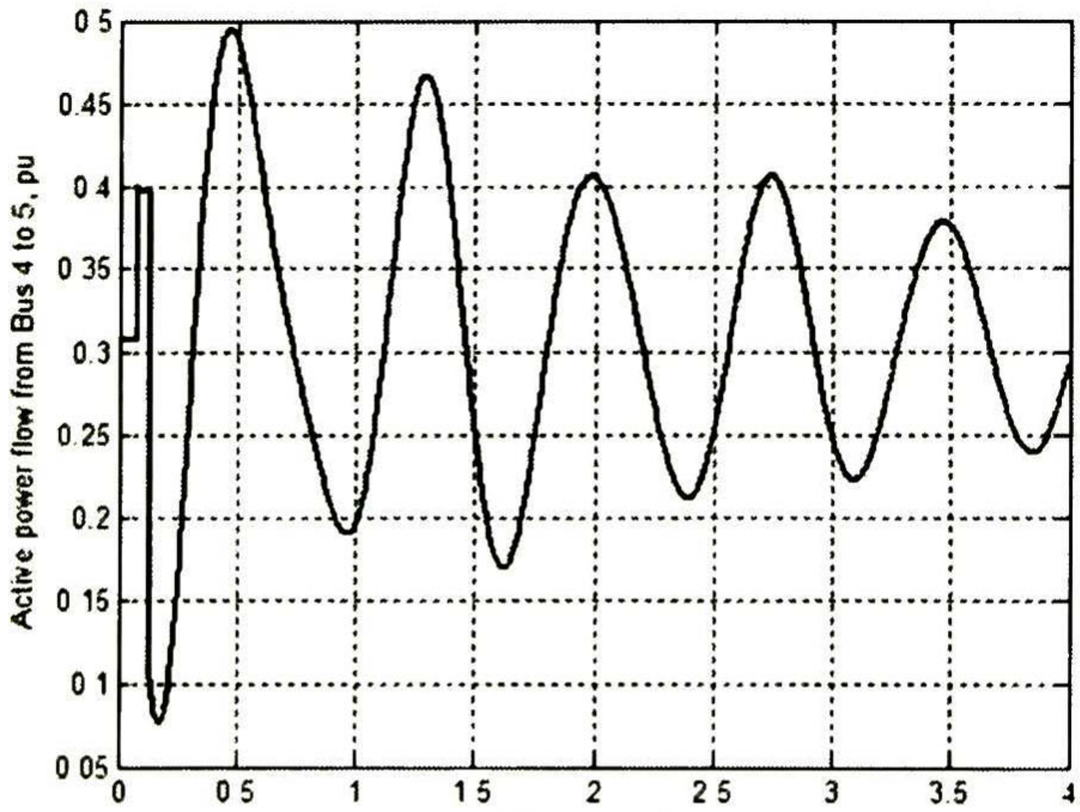



Fig. 4.18a Active power flow from Bus 4 to 5 under a three-phase fault at node 5.

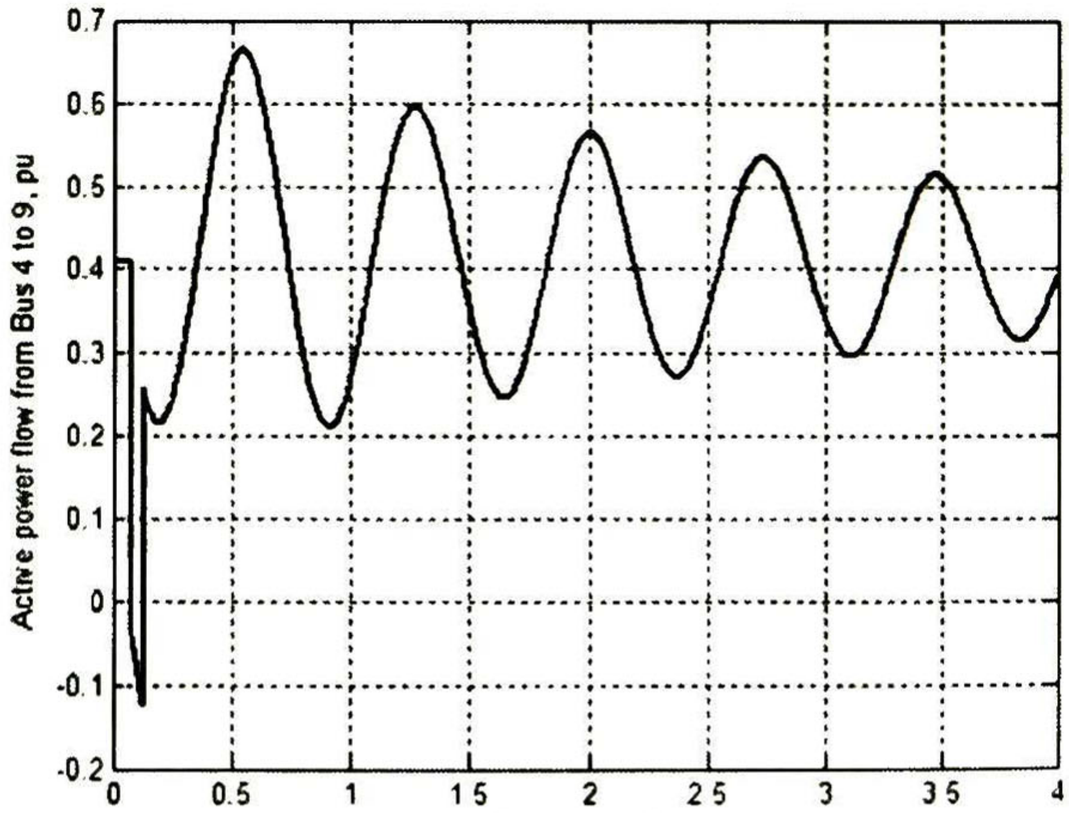



Fig. 4.18b Active power flow from Bus 4 to 9 under a three-phase fault at node 5.

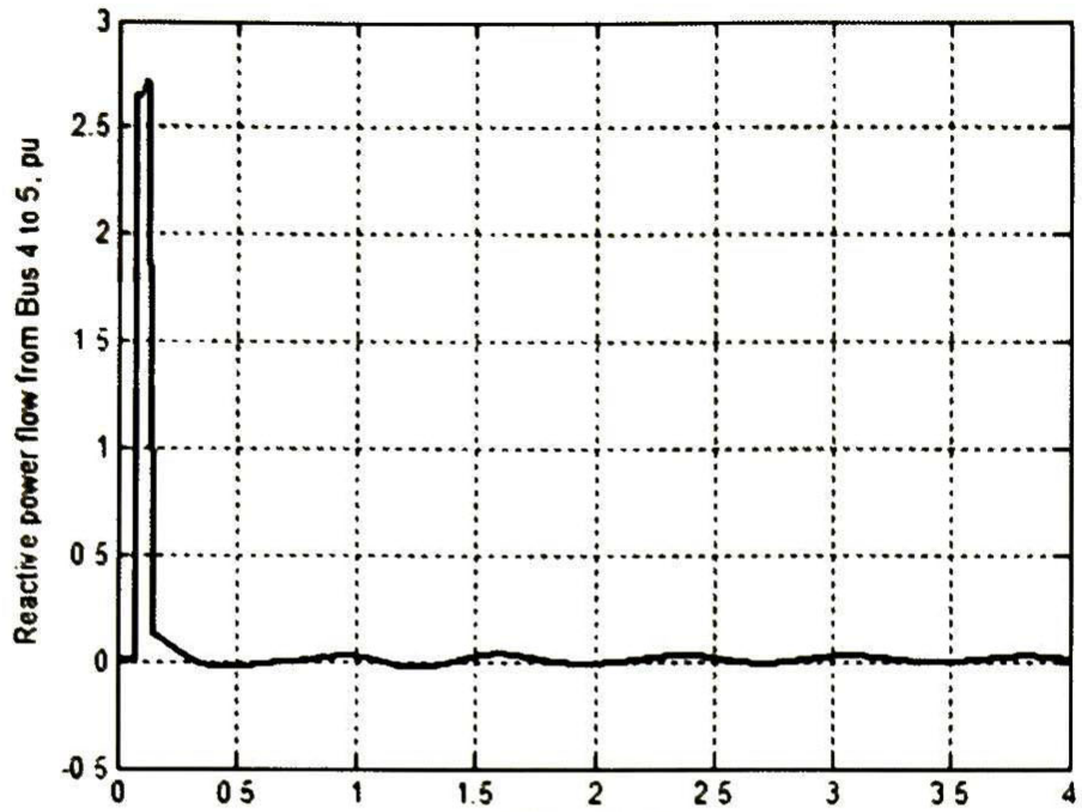



Fig. 4.19a Reactive power flow from Bus 4 to 5 under a three-phase fault at node 5.

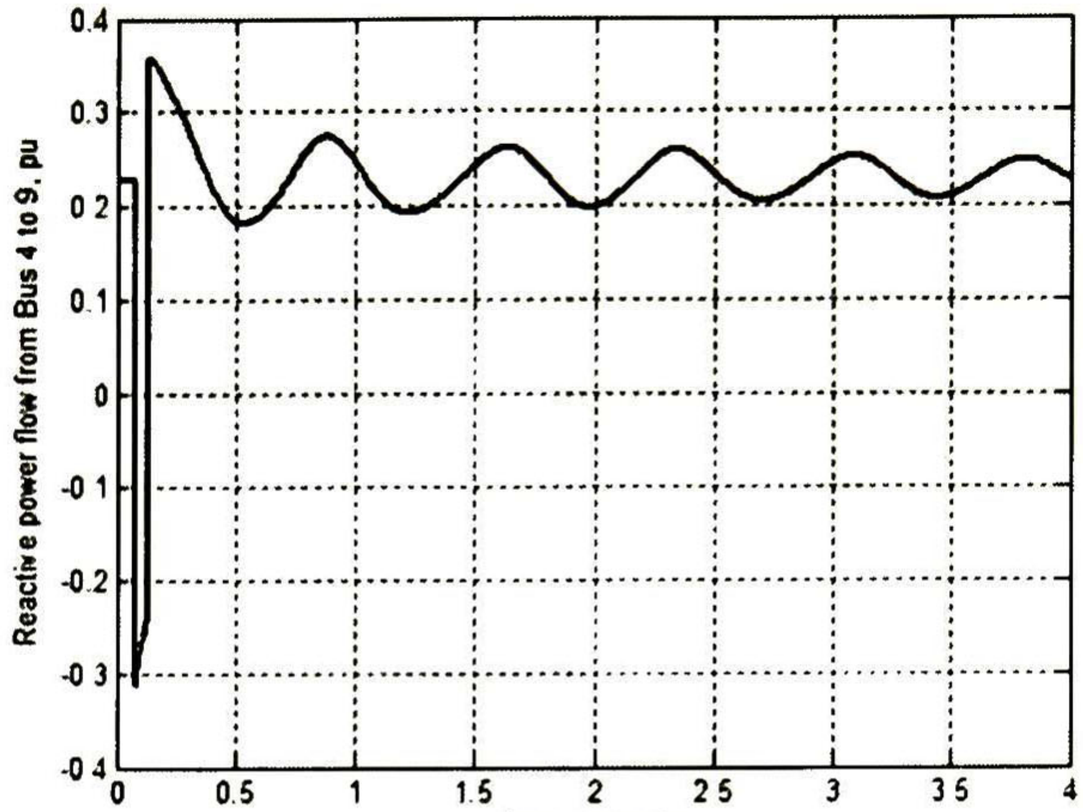



Fig. 4.19b Reactive power flow from Bus 4 to 9 under a three-phase fault at node 5.

On purpose, all data are trained and it is done for the complete simulation time (4 seconds). After that the prediction is done. Such prediction consists in foreseeing the voltage at Bus 4 when other disturbances are applied. These other ones are: (1) a load variation at Bus 7 and (2) the line that connect Bus 5 to Bus 6 is tripped. Permit us explain the load variation with more details. In this disturbances, both the active and reactive powers that are connected at Bus 7 are reduced 35%.The studied time is 4 seconds. Figs. 4.20 and Figs. 4.21 show the machine's performance under these other disturbances.

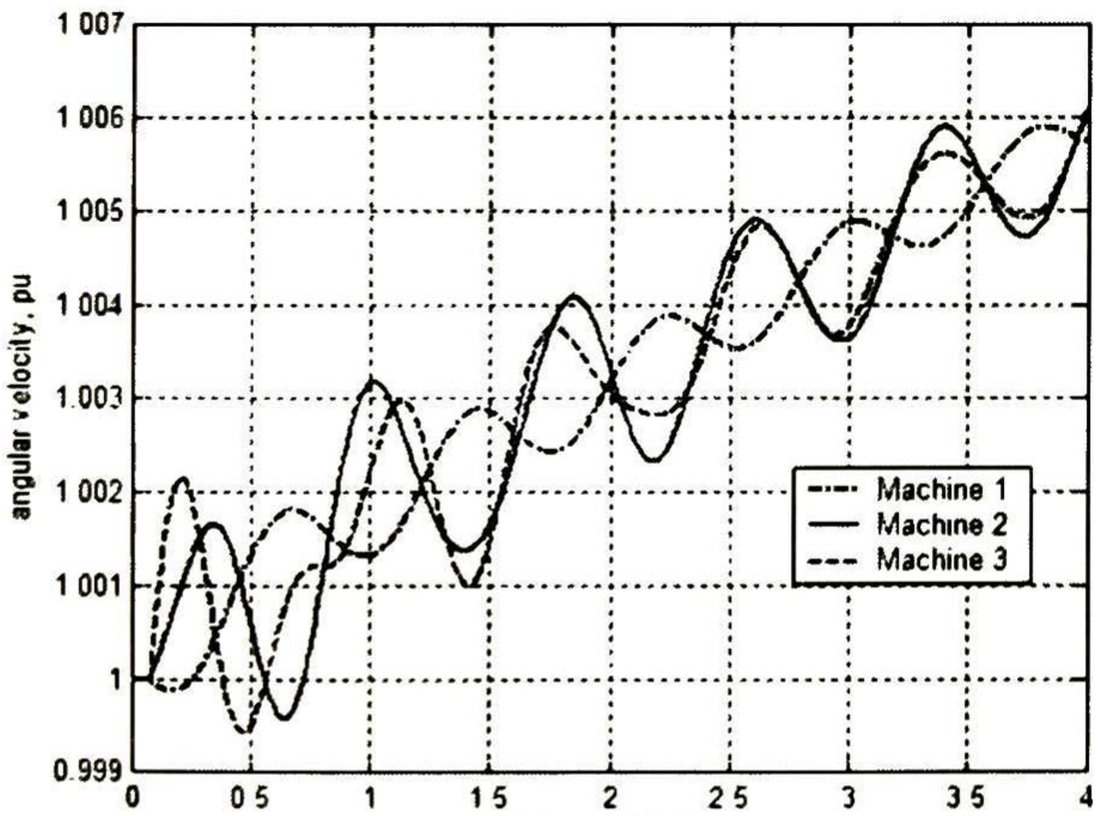



Fig. 4.20a Angular velocity performance when line 5-6 is tripped.

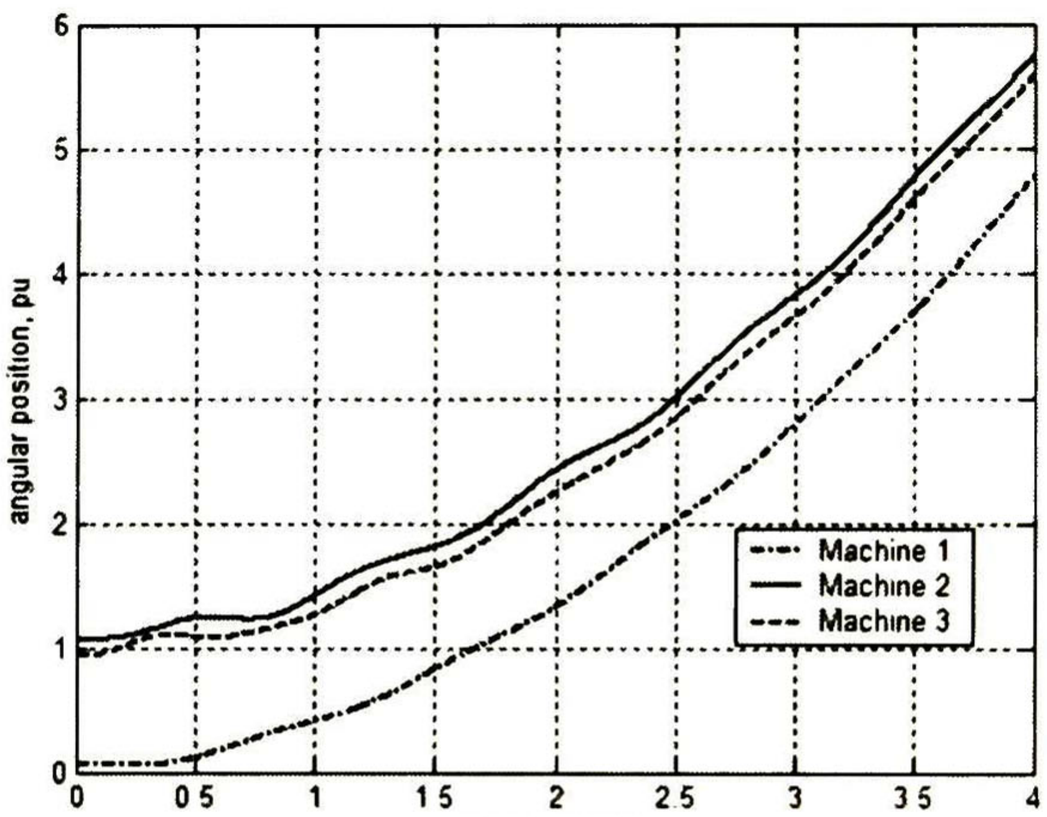



Fig. 4.20b Rotor machine performance when line 5-6 is tripped.

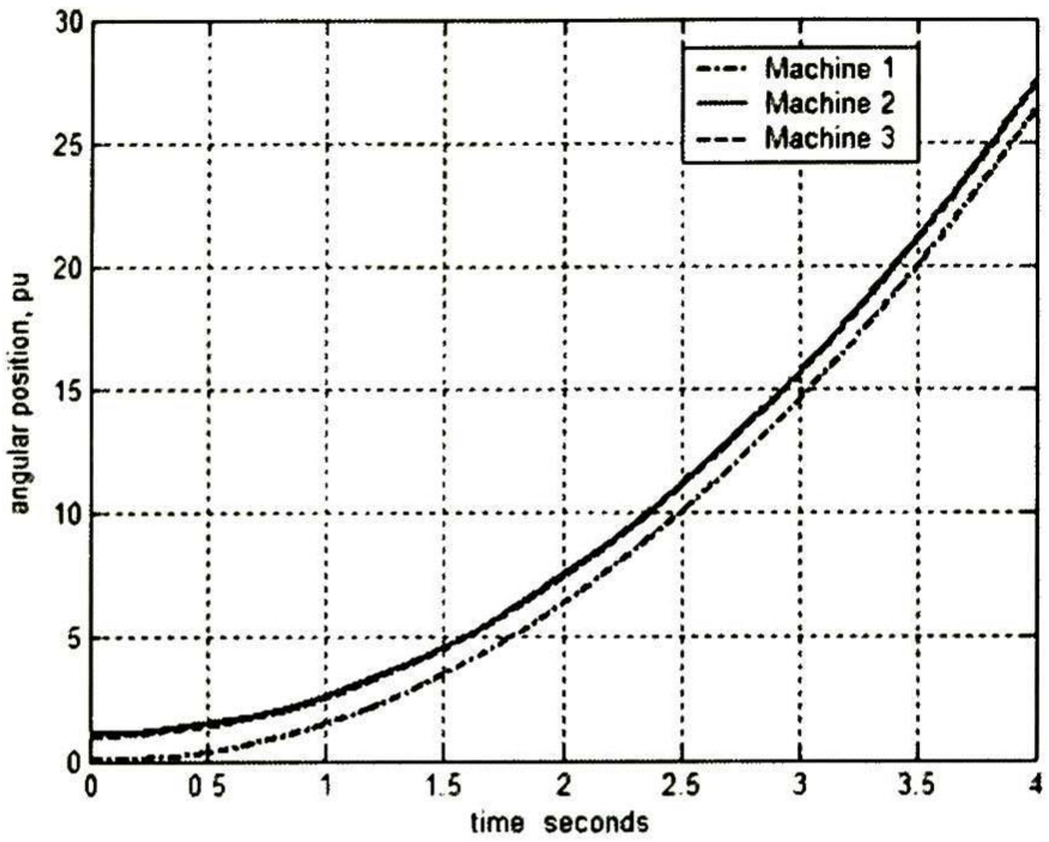



Fig. 4.21a Angular position performance when a load variation is presented at Bus 7.

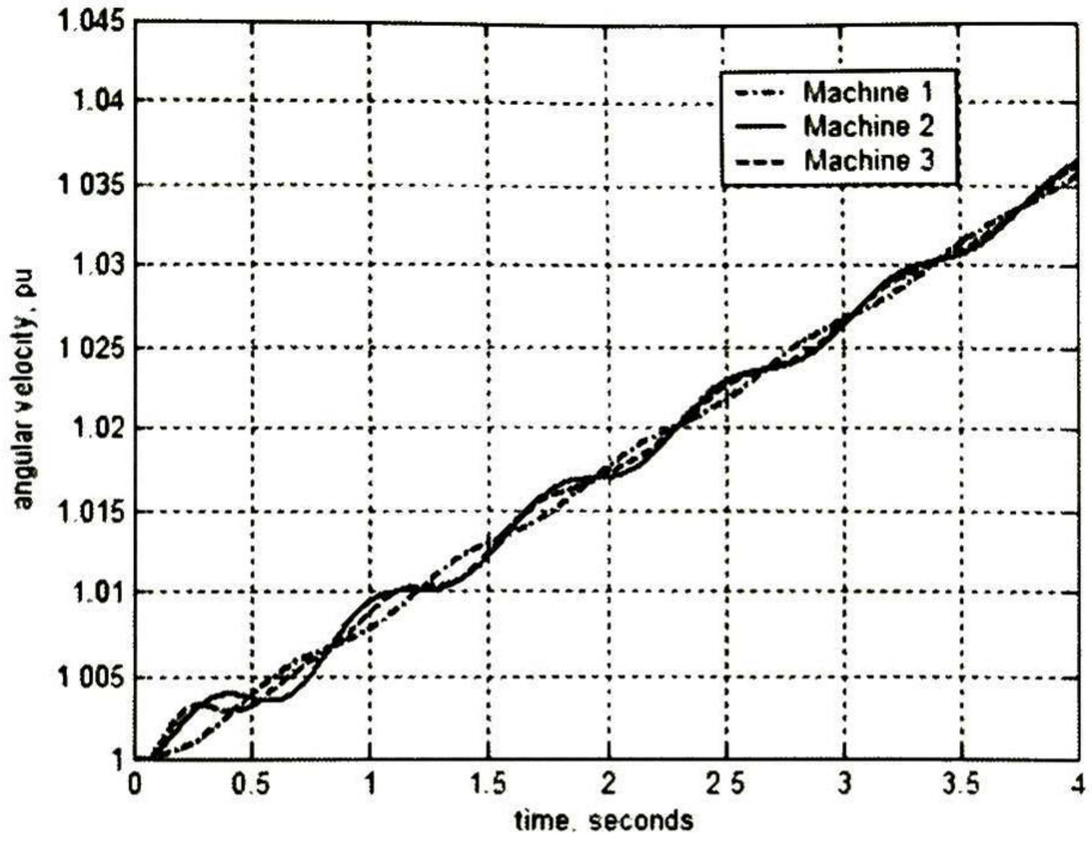



Fig. 4.21b Rotor machine's performance when a load variation is presented at Bus 7.

Figs. 4.22 and 4.23 depict the comparison between the prediction done and the actual results; the actual results are the simulation outputs from a transient stability programme. The dashed lines represent the prediction results made by the artificial neural network and the solid lines represent the simulation outputs.

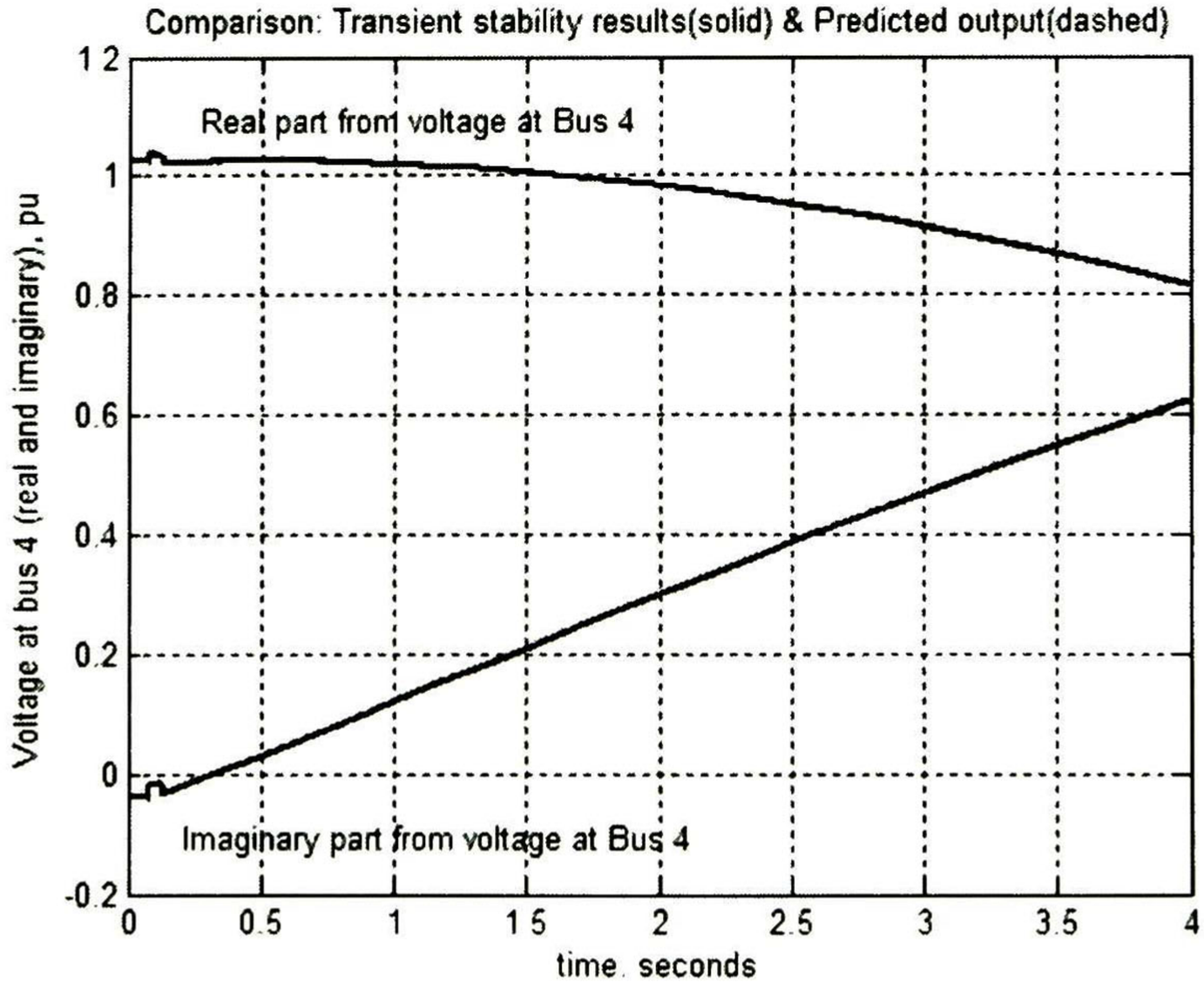



Fig. 4.22 Voltage behaviour at Bus 4 when a load variation is presented.

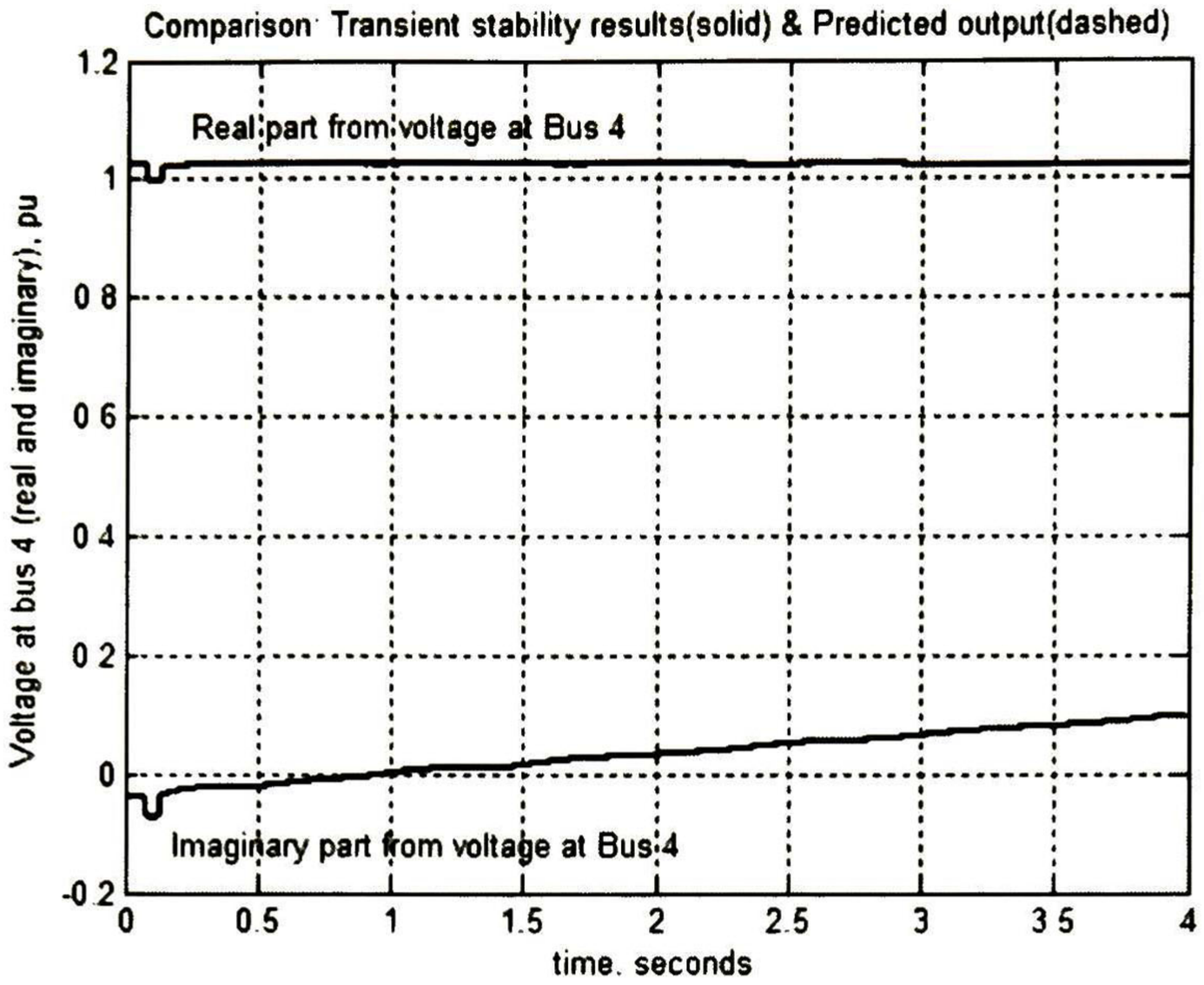



Fig. 4.23 Voltage behaviour at Bus 4 when line 5-6 is tripped.

As it is noticed, the neural network used in this example present great closeness between real and predicted values.

Section 4.3 and 4.3.1 show that a NN can be trained to predict some terminal voltages. The appropriated performance of this signal is believed to be fundamental for good dynamic equivalent.

## 4.4 Application of ANN to develop Dynamic Equivalents.

The second stage of this application consists in replacing the second machine, Fig. 4.15, -including the transformer and Bus 2- by a dynamic equivalent which it possesses only the same inertia as the replaced one. The equivalent machine is represented by a second order model; this is done just for simplicity. A neural network is employed to predict the complex voltage at node 8.

### 4.4.1 The Artificial Neural Network.

A feedforward neural network is used for determining the terminal voltage $V_8$. The neural network is trained taking as input values the active and reactive (real and imaginary) power flows from several disturbances. The input values are the active and reactive power flows from Buses 8-7 and 8-9 of some faults applied to the complete system (Fig. 4.15), and the output values are considered as the complex voltage at Bus 8; since this voltage is a complex number, so akin to others ANNs that have been already trained, it is necessary to create two neural networks, one to get the real value and a second one for the imaginary part. The training procedure for obtaining the real and the imaginary values is done with the structure showed in Fig. 4.24.

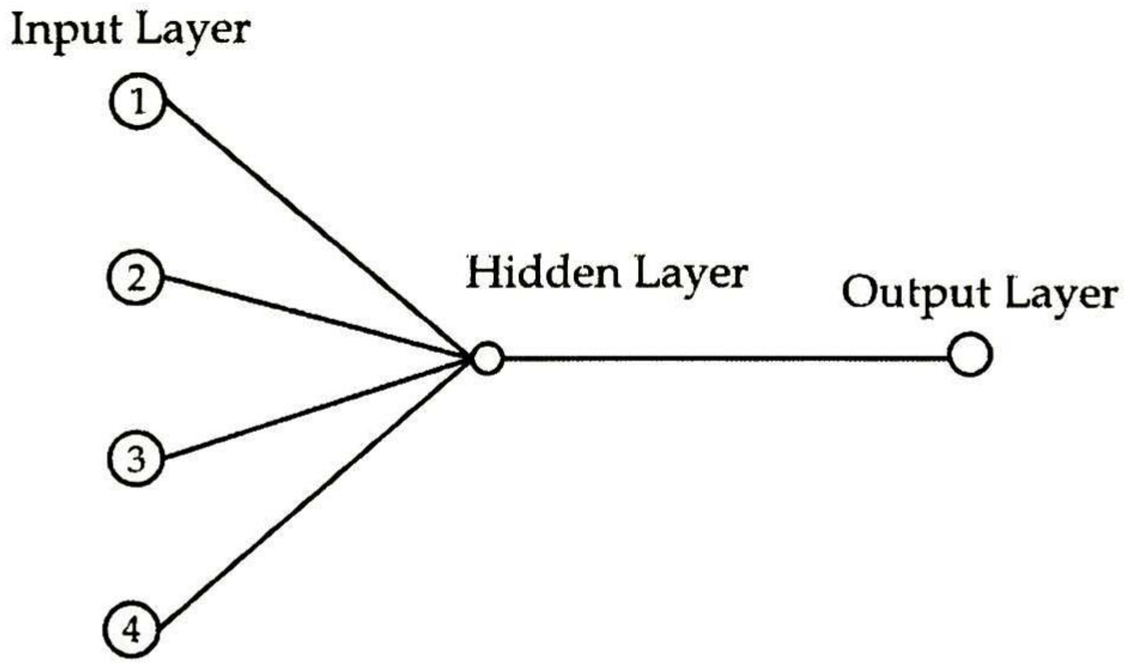



Fig. 4.24 ANN structure.

This structure is selected after considering many other configurations and it presents the best results; the hidden layer is activated by a sigmoid function (tangent-hyperbolic transfer function) while the output layer is activated by a linear function. The training is done for 1000 epochs but, it always converges before the epoch number is reached.

To get the greatest trained data, during the training algorithm some processes are done. Before the data are trained all the values are scaled; this is done with the aim to remove the mean and scale all signals to the same variance, for this case the variance is taken as one and the mean is zero. If the data are not scaled, the largest values tend to be dominant. After the training process finishes the data need to be rescaled so, the network model can be used for any purpose.

As we already stated, the input values for training purposes are the active and reactive power flows from Buses 8-7 and 8-9 from some disturbances applied to the complete system (Fig.4.15), and the output values are considered as the complex voltage at Bus 8. The input values are obtained from a transient stability simulation applying several disturbances to the system in the following manner. The first disturbances consist in a variation of the line parameters; for this one (fault 1) the line affected is the one which connects Bus 5 to Bus 6 and the total simulation time is 1s; the change is applied at 0.50s and it is never cleared.

Table 4.9 Transmission lines parameters for disturbances purposes.

| TRANSMISSION LINES PARAMETERS | | | | | |
|---|---|---|---|---|---|
| From Bus No | To bus No | Resistance (pu) | Reactance (pu) | Line charging (pu) | Tap ratio |
| 5 | 6 | 0.0 | 0.50 | 0.0 | 1.0 |
| 7 | 8 | 0.0 | 0.50 | 0.0 | 1.0 |
| 4 | 9 | 0.0 | 0.50 | 0.0 | 1.0 |

The same process is done for lines that connect Bus 7 to Bus 8 (fault 2), and the line that connect Bus 4 with Bus 9 (fault 3). Table 4.9 shows the modified transmission line parameters applied to get these

three faults. Fig. 4.25, 4.26 and 4.27 are examples of these faults, and Fig. 4.28 and 4.29 illustrate some of the input signals that are used to train the neural network.

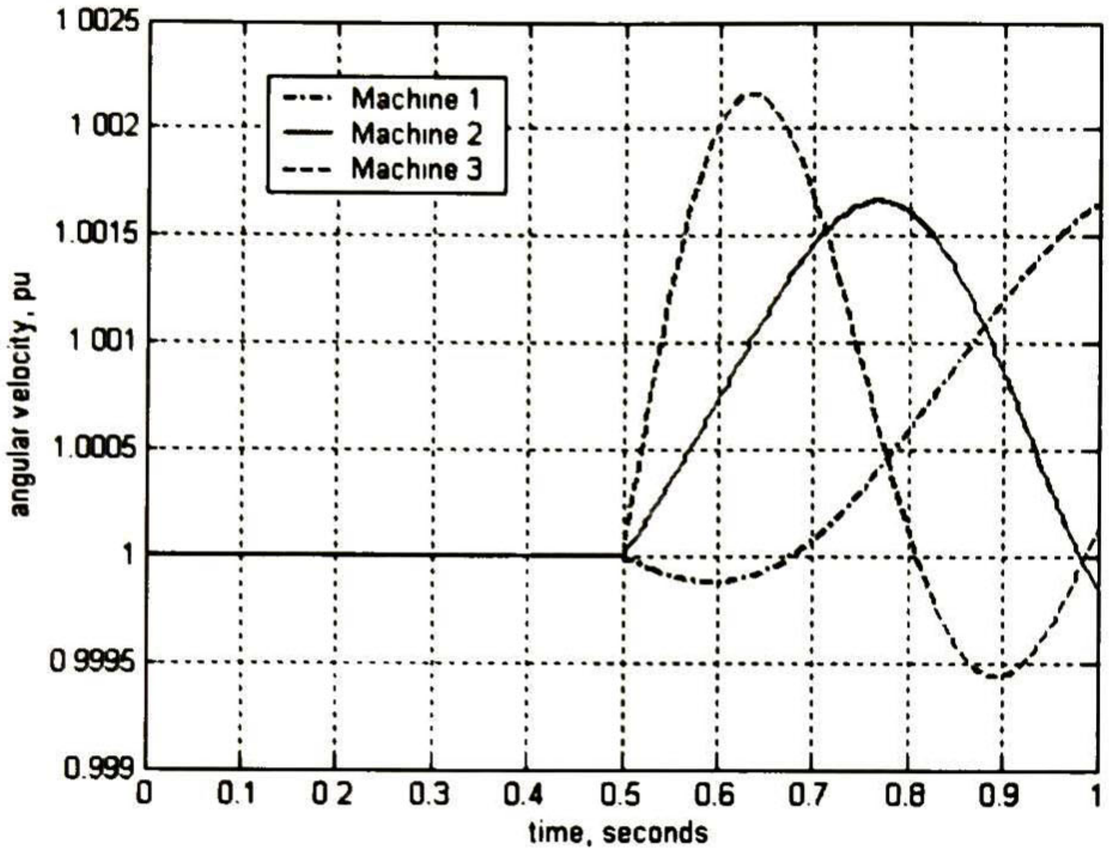



Fig. 4.25 Angular velocities under fault 1.

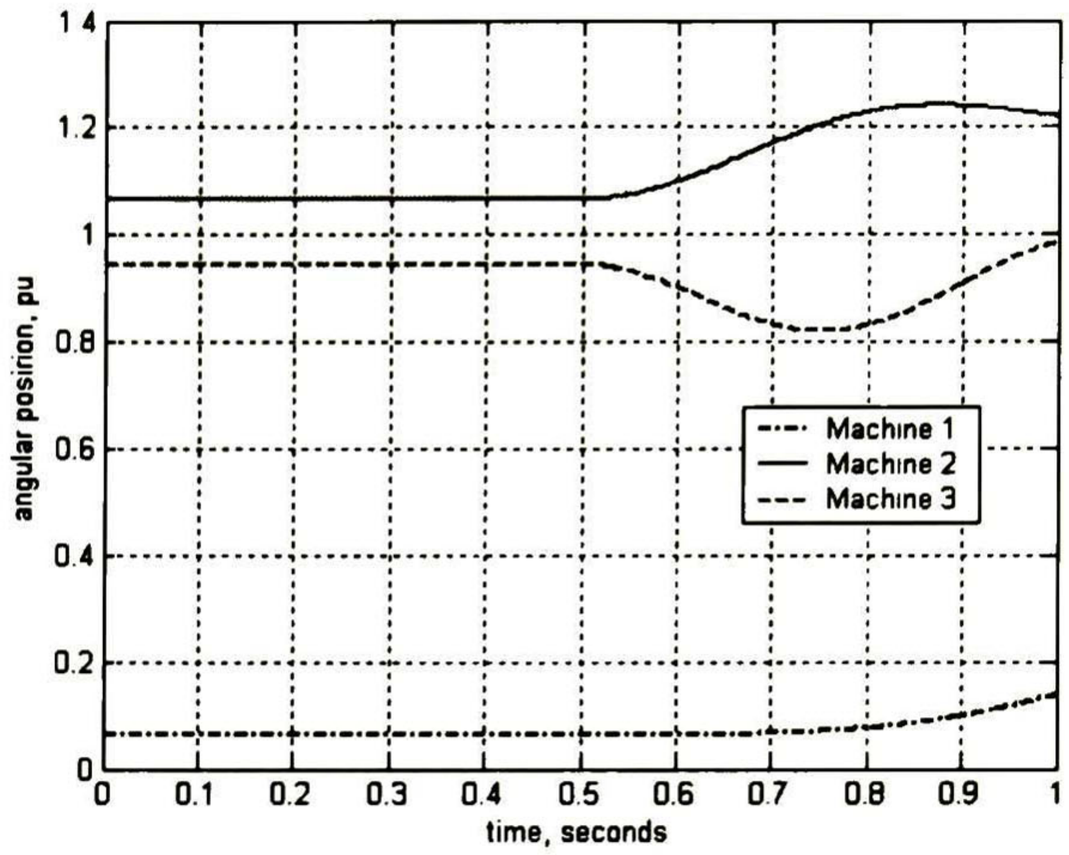



Fig. 4.26 Rotor machine's under fault 2.

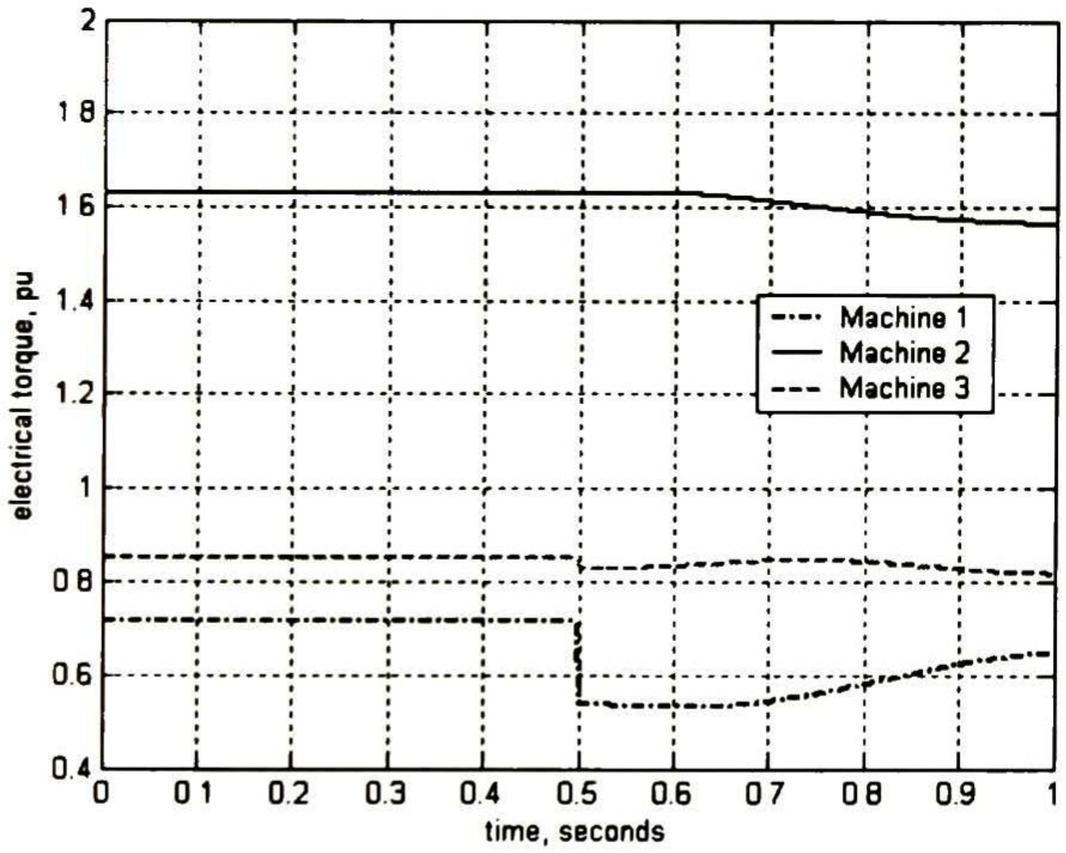



Fig. 4.27 Electrical torque under fault 3.

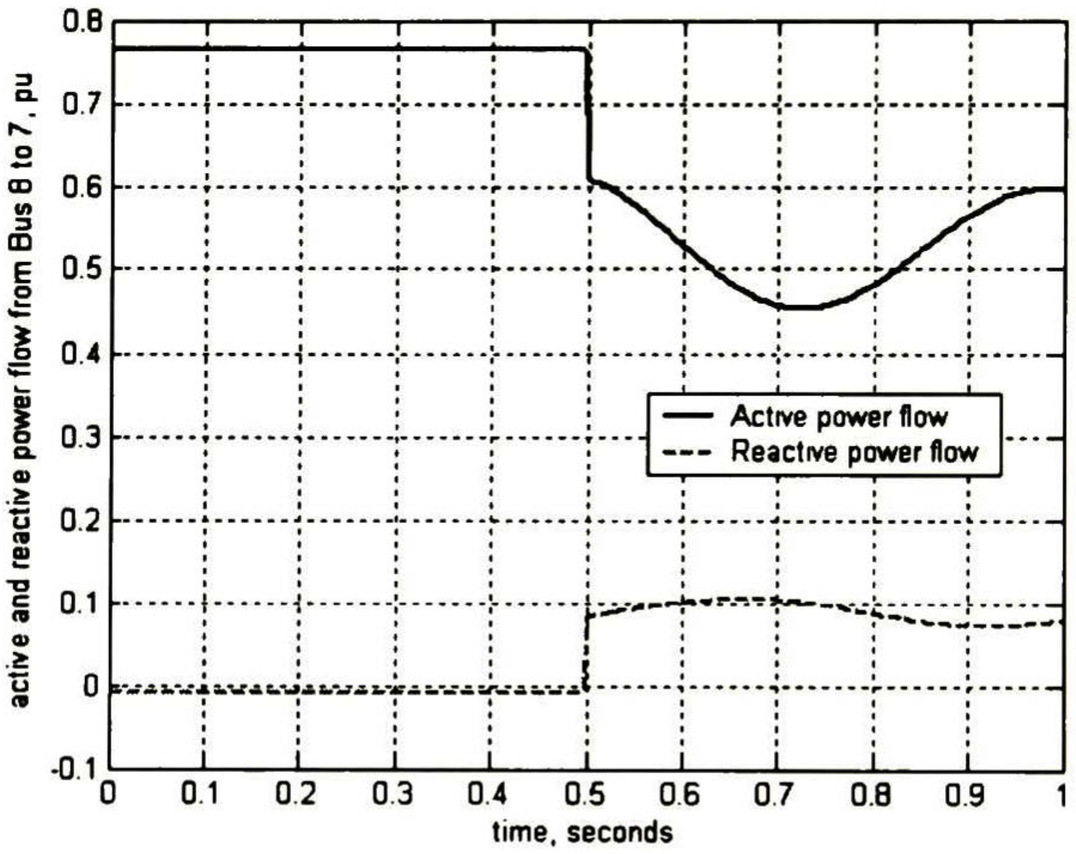



Fig. 4.28 Active and reactive power flow from Bus 8 to 7 under fault 1.

The subsequent disturbances are a variation of power load at Buses 5, 7 and 9. The disturbance is performed turning up or down the power loads; for Bus 5 the variation is a decrement of load (both, active and reactive) and like in all cases it is applied at 0.50s from the starting time; for Bus 7 power load is increased; the third load variation which is done at Bus 9 is also an increment of active and reactive power load at that Bus. Like in the former cases the load variations are never cleared and the total simulation time is for 1s.

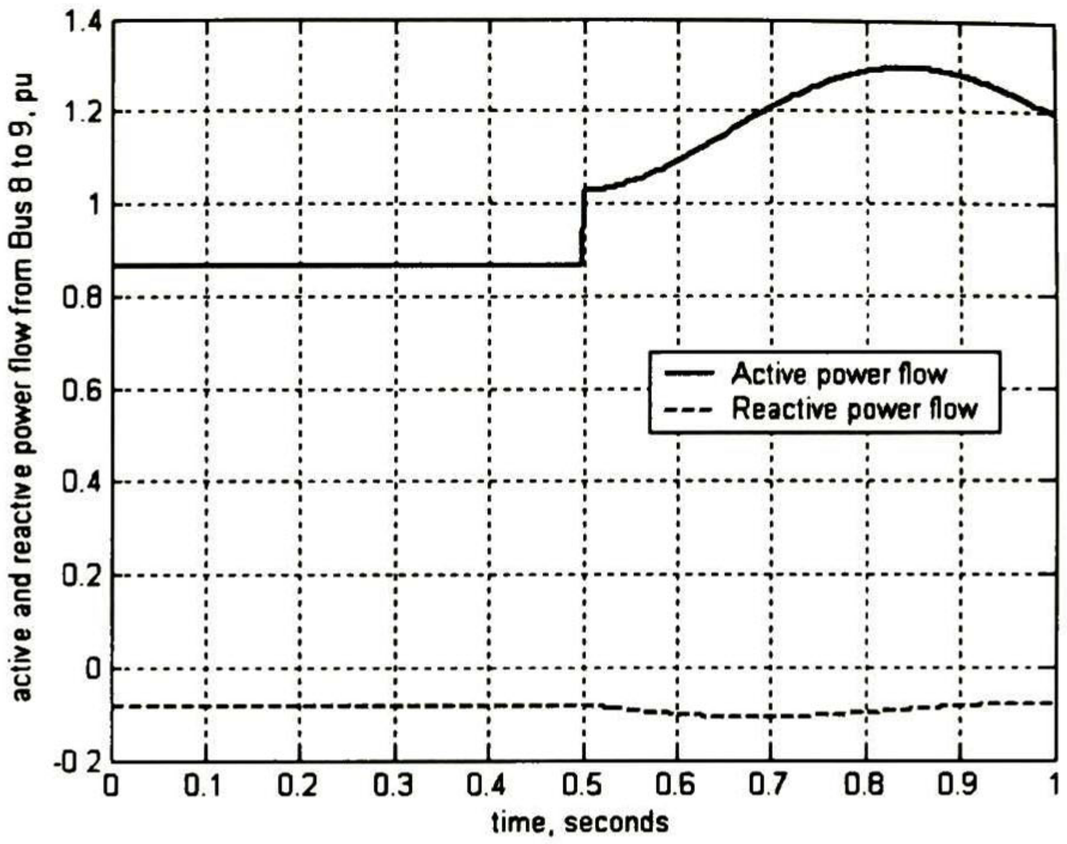



Fig. 4.29 Active and reactive power flow from Bus 8 to 9 under fault 2.

To be convinced that ANN is well trained, it is proved or validated. The validation is done considering many different faults from the ones that are used to be trained. The procedure is almost the same as in the training method but now, the disturbances are changed. In the validation, the most important values that we need to check are the weights and also we verify that the input values that are considered are the best ones. Figs. 4.30 depict one example of how the validation is done considering a three phase fault at Bus 5.

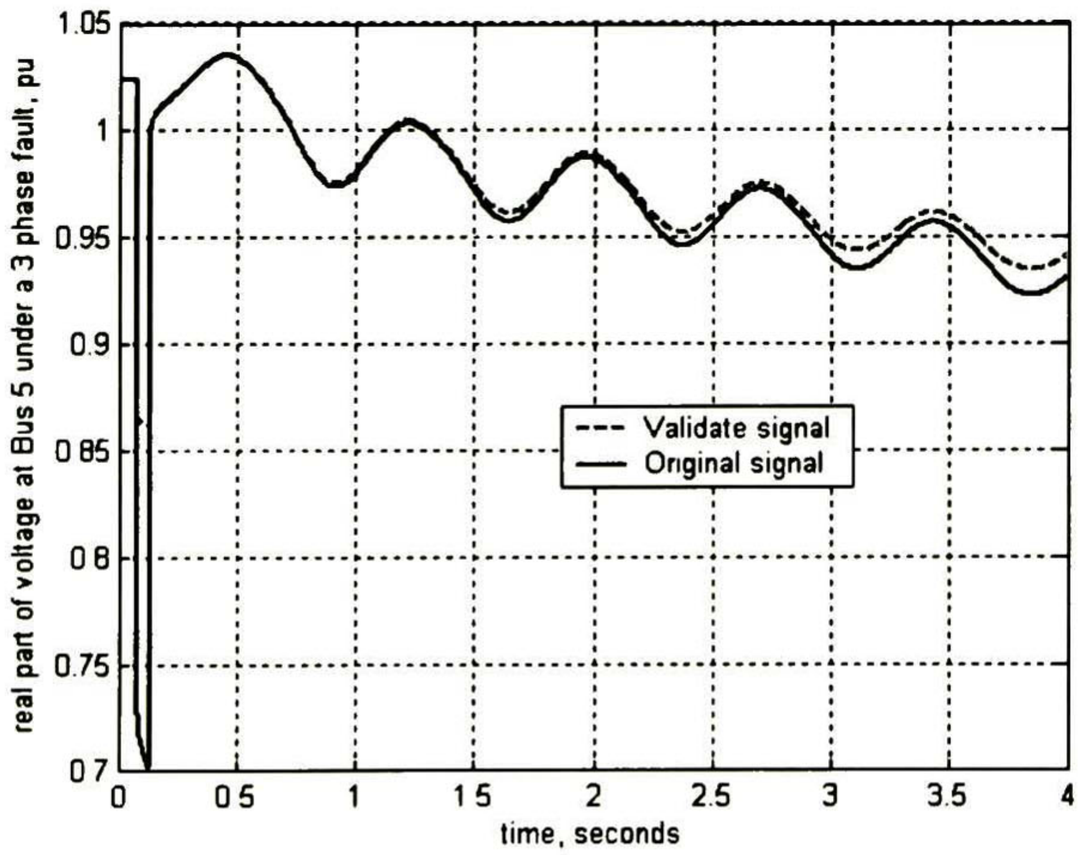



Fig. 4.30a Validation of the real part of voltage after training.

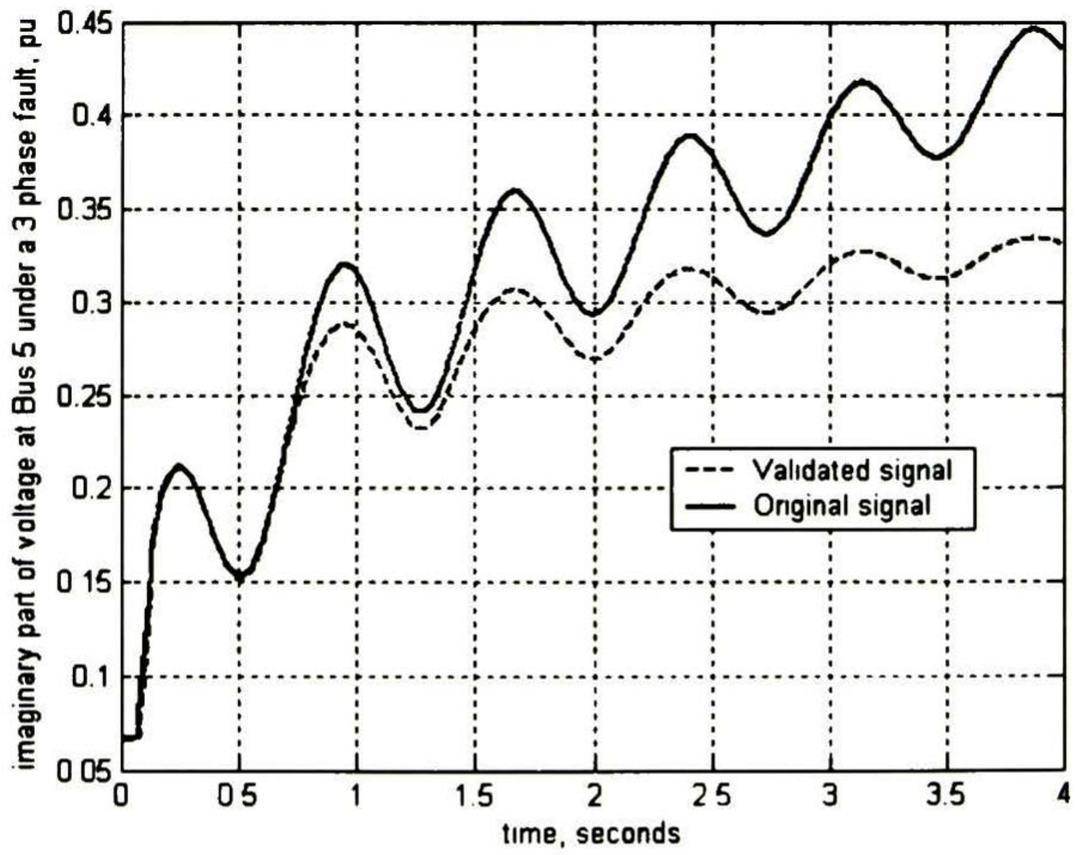



Fig. 4.30b Validation of the imaginary part of voltage after training

Afterwards the training and the validation is done, the main objective is completed. At this stage a transient stability programme is executed. This transient stability study has some differences from a normal study; in other words, the transient stability programme is combined with a neural network, to can reach our main objective. Fig. 4.31.

The power system configuration in Fig. 4.15 is replaced by the system in Fig. 4.32 and the forecast is performed as it is described in Fig. 4.31.

The estimation is done employing an ARX model which is a MISO (multiple inputs - single output) ARX system. The voltage prediction at Bus 2 of the reduced system (Fig. 4.32) requires only 1 past input and 1 past output, and the prediction time delay is zero.

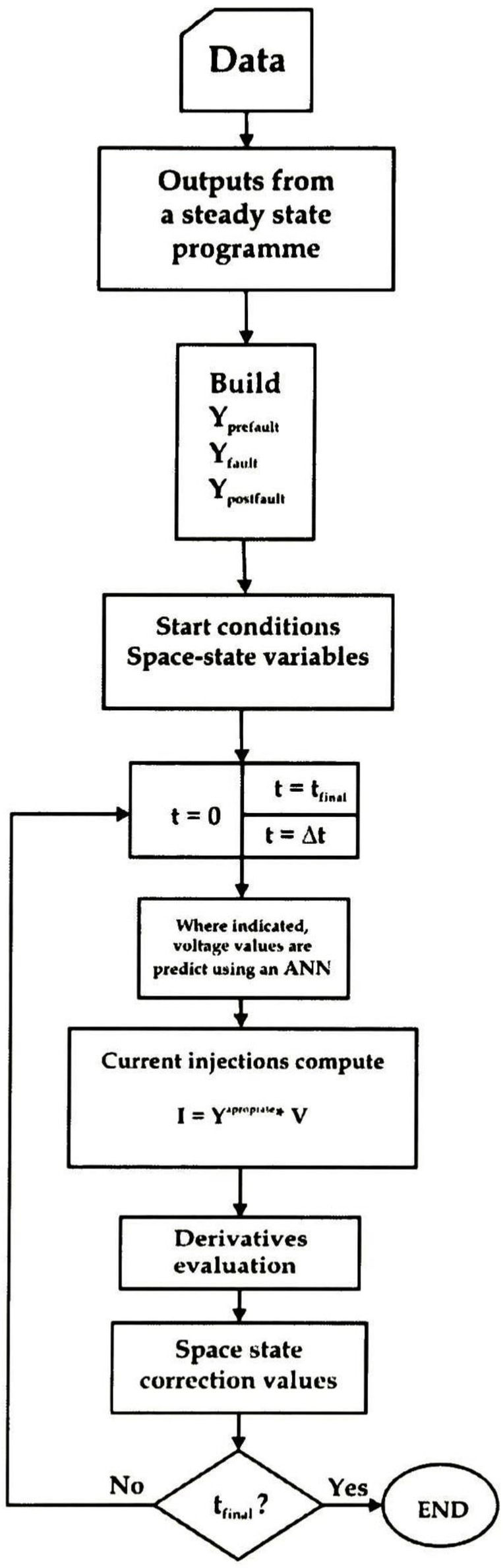



Fig. 4.31 Modified flow chart from a transient stability study.

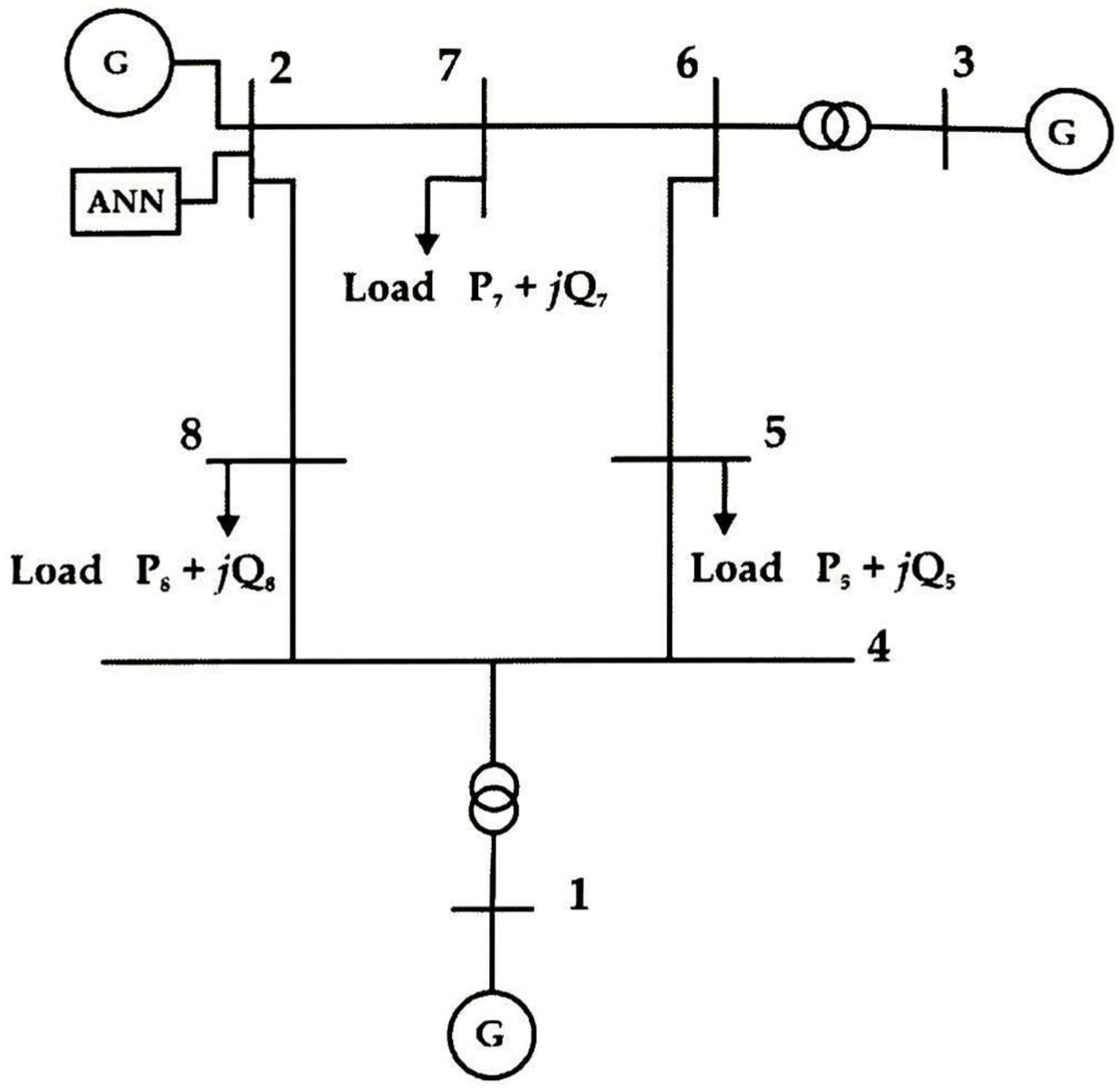



Fig. 4.32 3 Machines 9 buses reduced system.

Figs. 4.33, 4.34 and 4.35 depict the comparison between the prediction done and the actual results for the machine 1, and Fig. 4.36 shows the comparison between the prediction done and the actual results for the third machine; the actual results are the simulation outputs from a transient stability programme employing the base data of the complete system (Fig. 4.15), and the predicted values are obtained after the system has been reduced, substituting the second machine by an equivalent and an ANN that is able to reproduced the complex voltage. The dashed lines represent the predicted results.

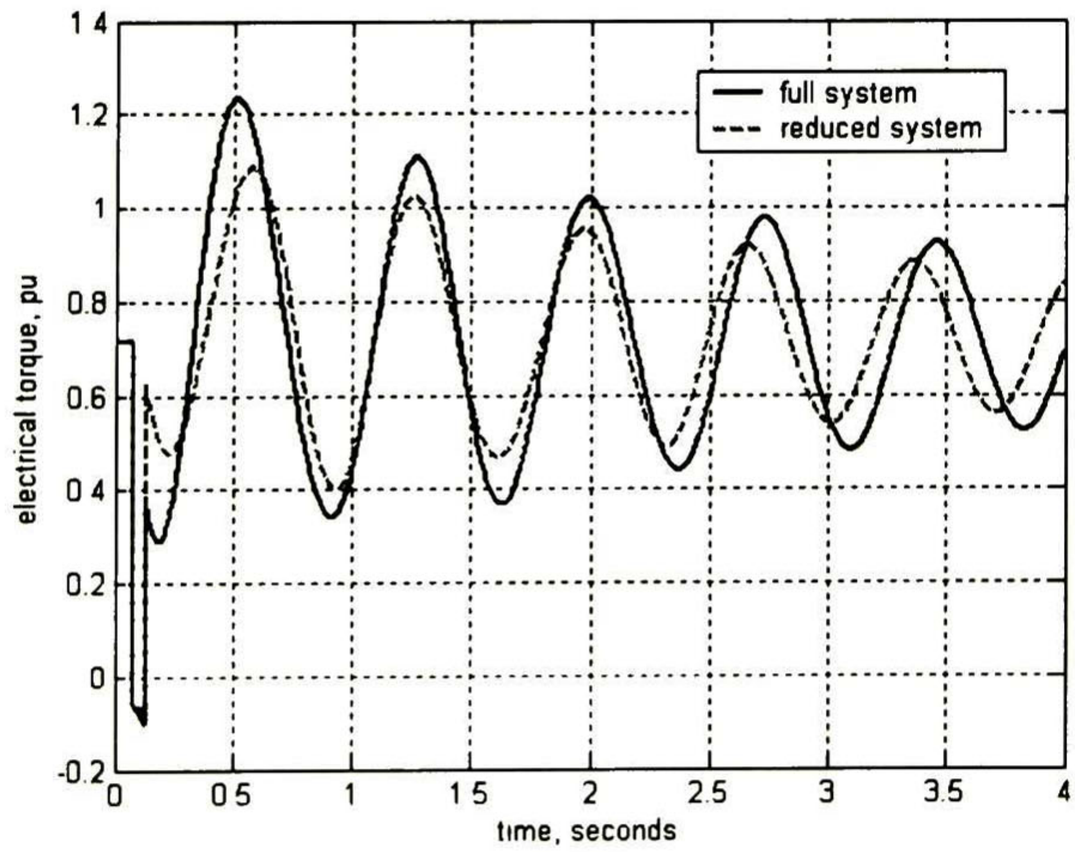



Fig. 4.33 Electrical torque from machine 1 under a three phase fault at Bus 4.

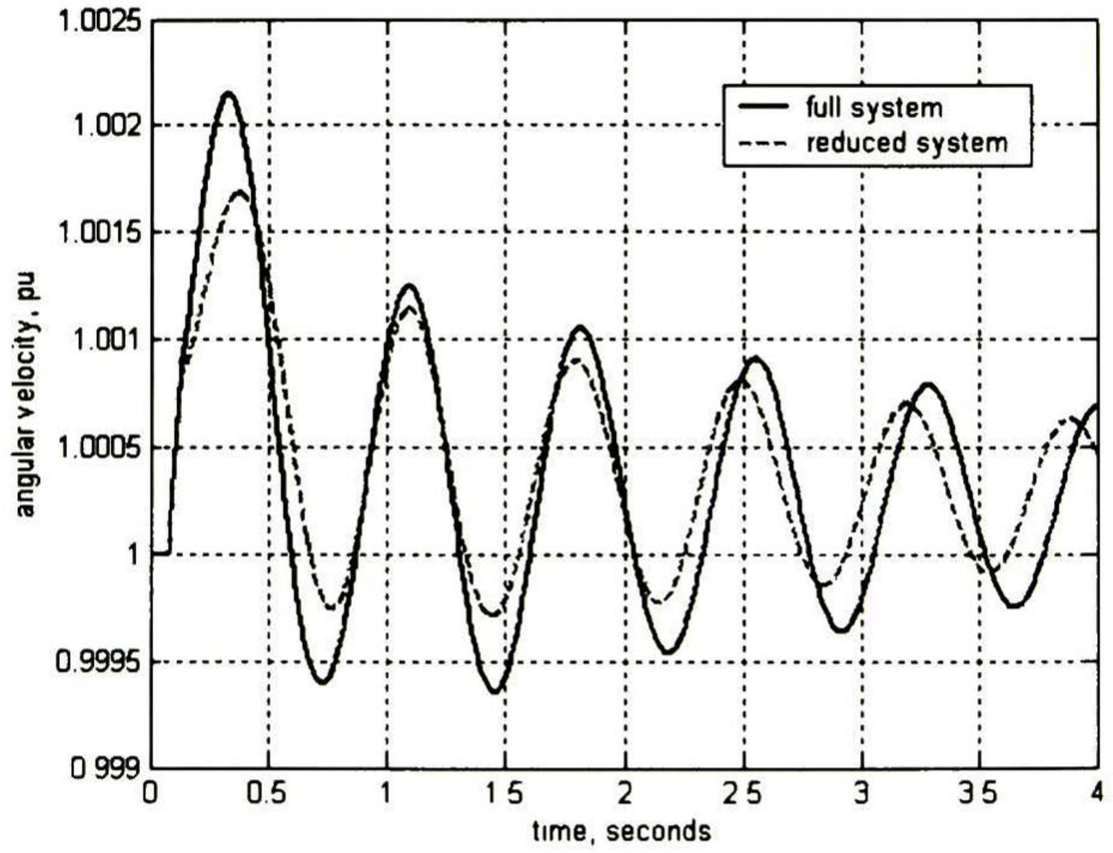



Fig. 4.34 Angular velocity behaviour from machine 1 under a three phase fault.

Fig. 4.37 shows the predicted voltage at Bus 2 from the reduced system (Fig. 4.32) which is the central purpose of this example. The dashed line represents the absolute voltage at Bus 2 from the reduced system; this value is forecasted during the transient stability study supported by the neural network, and the solid line represents the absolute voltage at Bus 8 from the complete system.

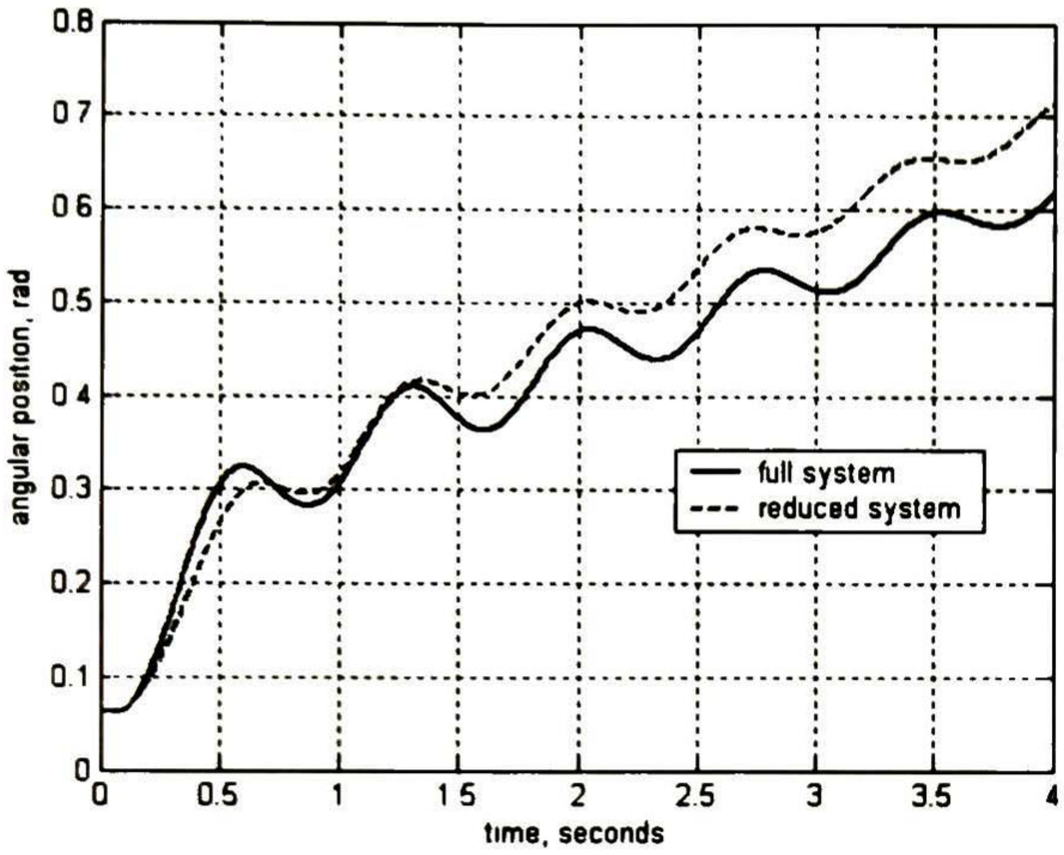



Fig. 4.35 Rotor machine's behaviour from machine 1 under a three phase fault at Bus 4.

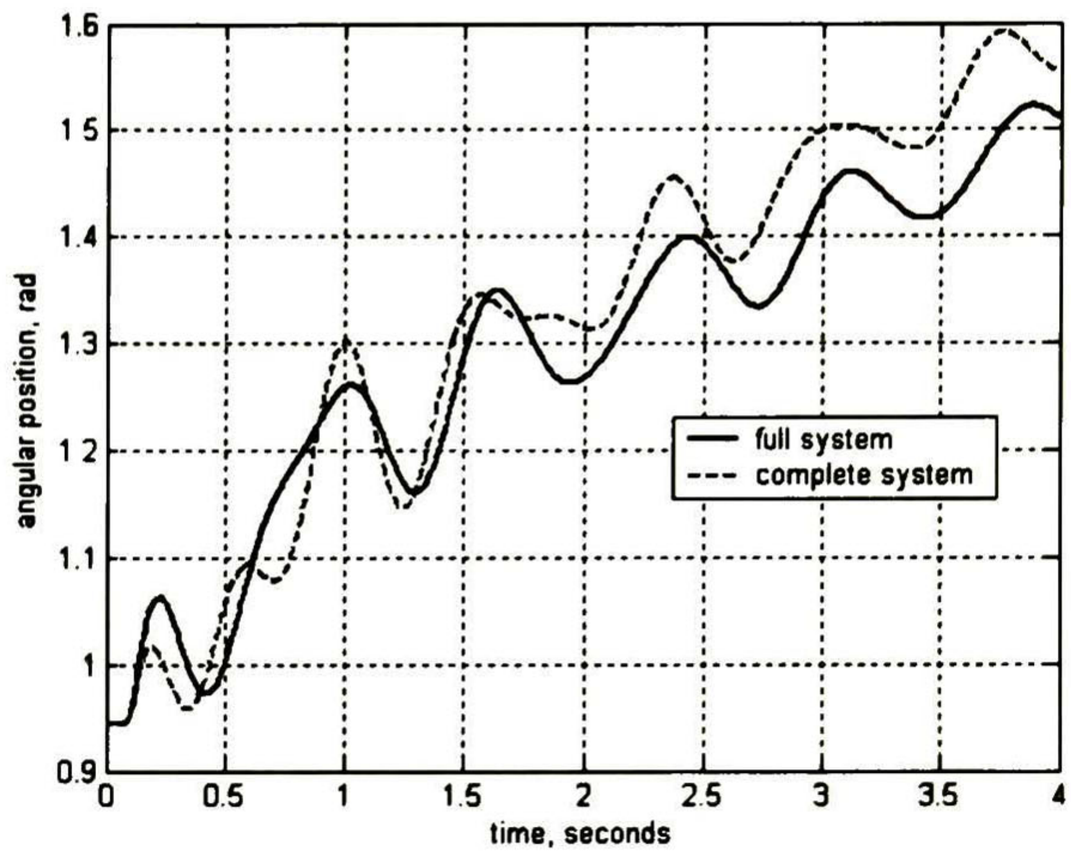



Fig. 4.36 Rotor machine's behaviour from machine 3 under a three phase fault at Bus 4.

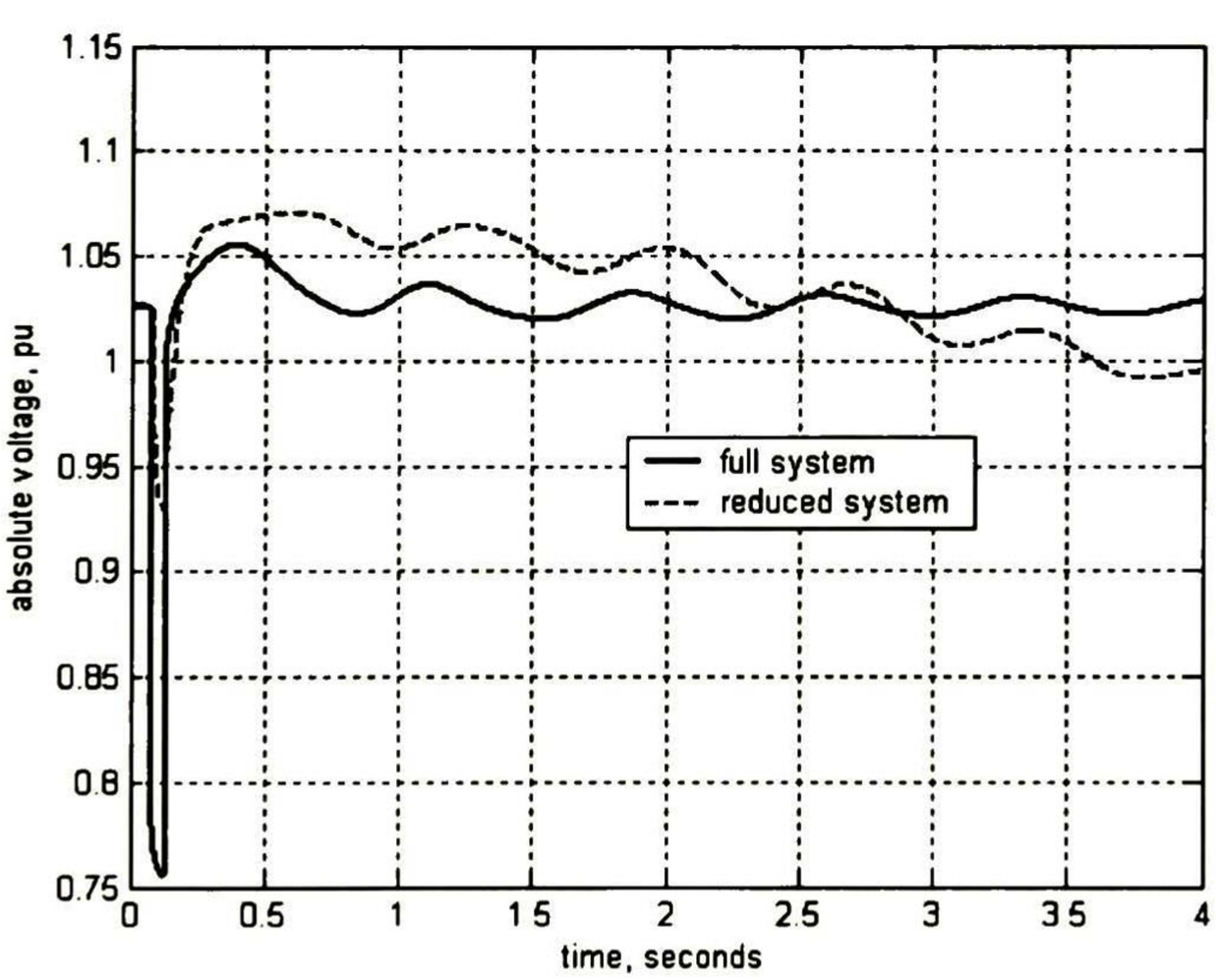



Fig. 4.37 Absolute voltage under a three phase fault.

In a similar manner, the reduced system is tested to prove the feasibility and certainty of the dynamic equivalent, under other disturbances. A load is added at Bus 6, which active power is defined to be 0.2 pu when the reactive power is established at 0.05 pu. Figs 4.38 and 4.39 describe the behaviour of the machine 1 of the reduced system under this disturbance, and Figs. 4.40 and 4.41 show the behaviour of the third machine. The solid lines represent the output obtained from the transient stability study

under this fault employing the data from the complete system, while the dashed lines correspond to the behaviour of the reduced system under the same perturbation.

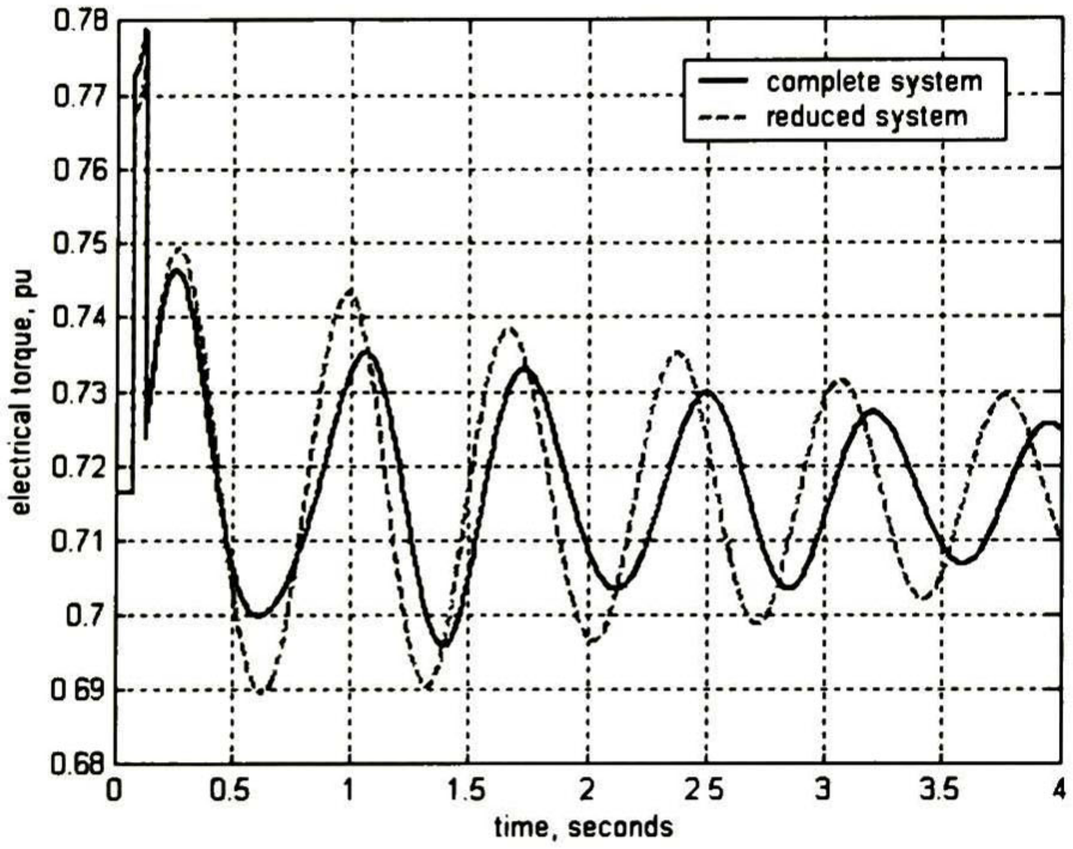



Fig. 4.38 Electrical torque from machine 1 when a load is presented at Bus 6.

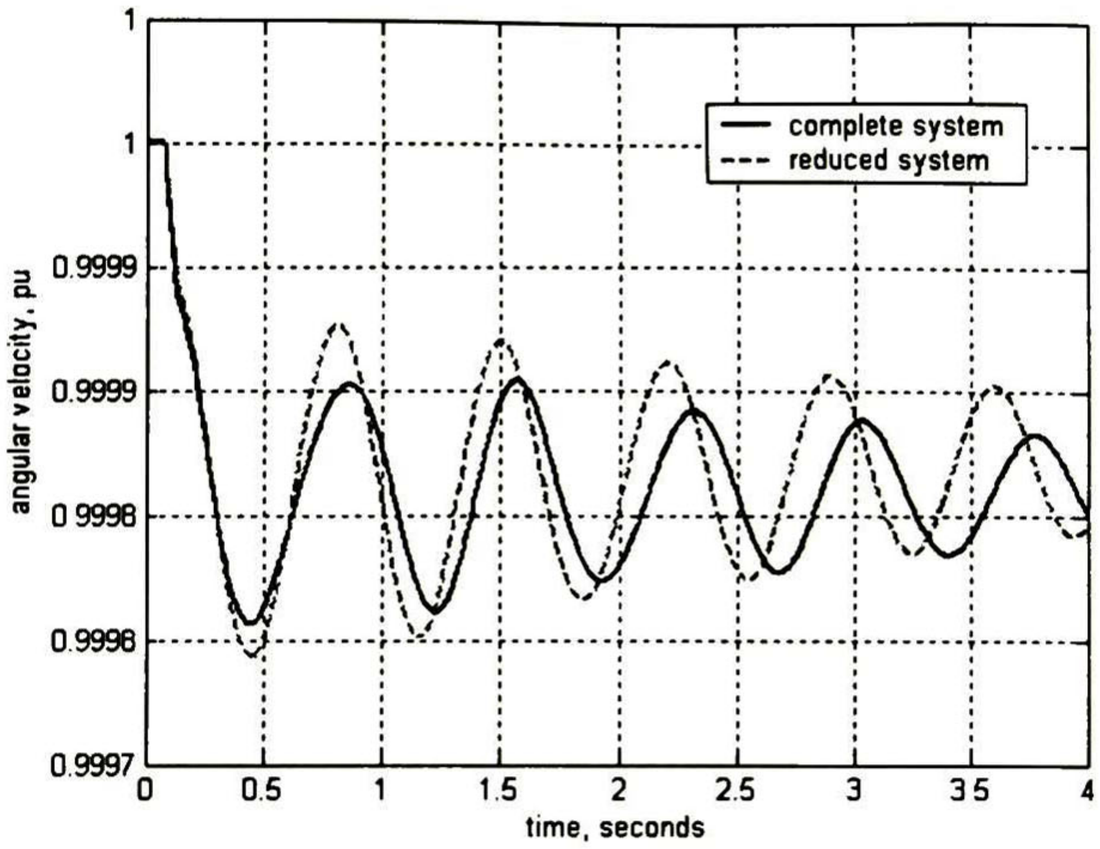



Fig. 4.39 Angular velocity behaviour from machine 1 when a load is presented at Bus 6.

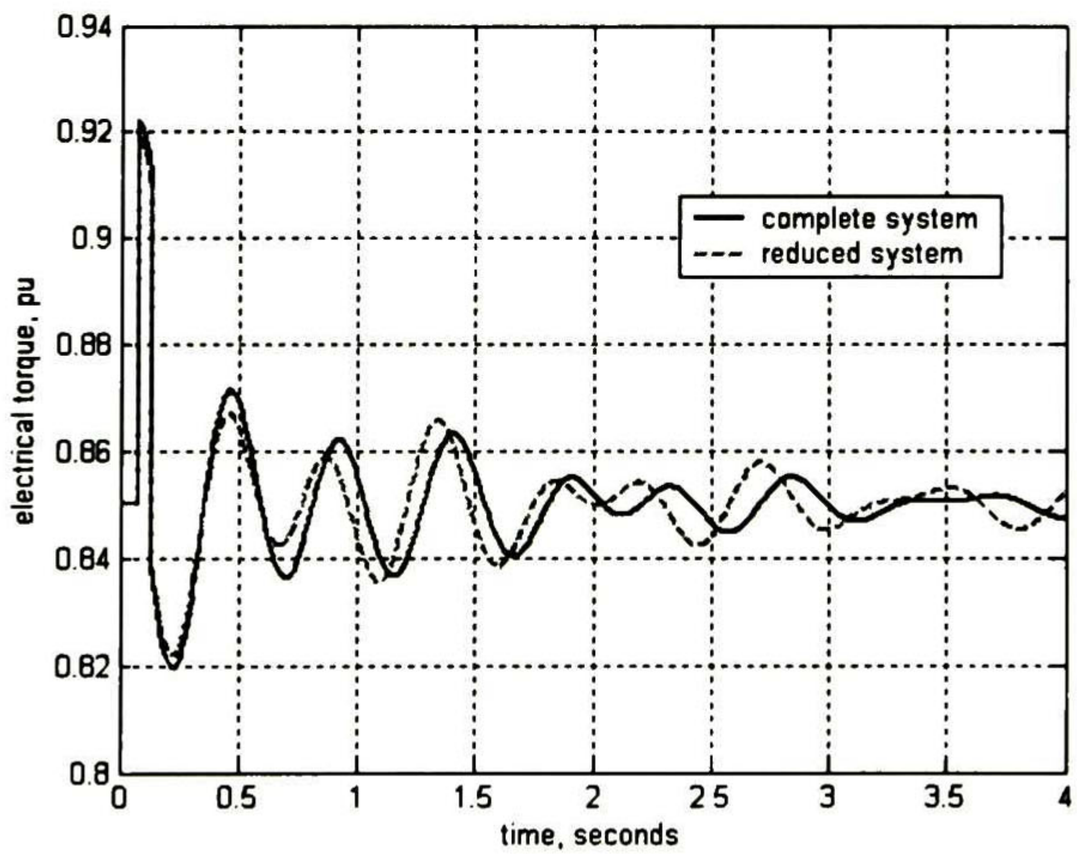



Fig. 4.40 Electrical torque from machine 3 when a load is presented at Bus 6.

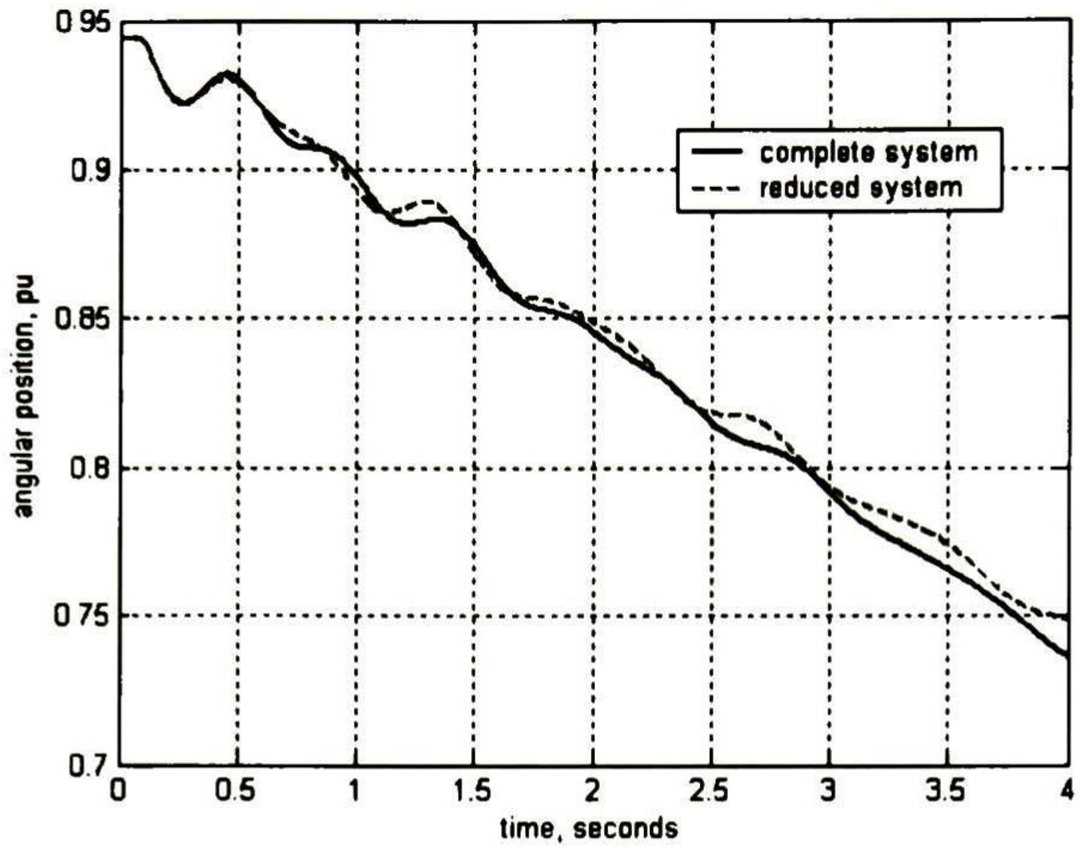



Fig. 4.41 Rotor machine's behaviour from machine 3 when a load is presented at Bus 6.

The second fault used to corroborate the warranty of the dynamic equivalent is the modification of the line parameters from Bus 3 to Bus 8. Figs. 4.42-4.45 depict the behaviour among the reduced system and the complete one.

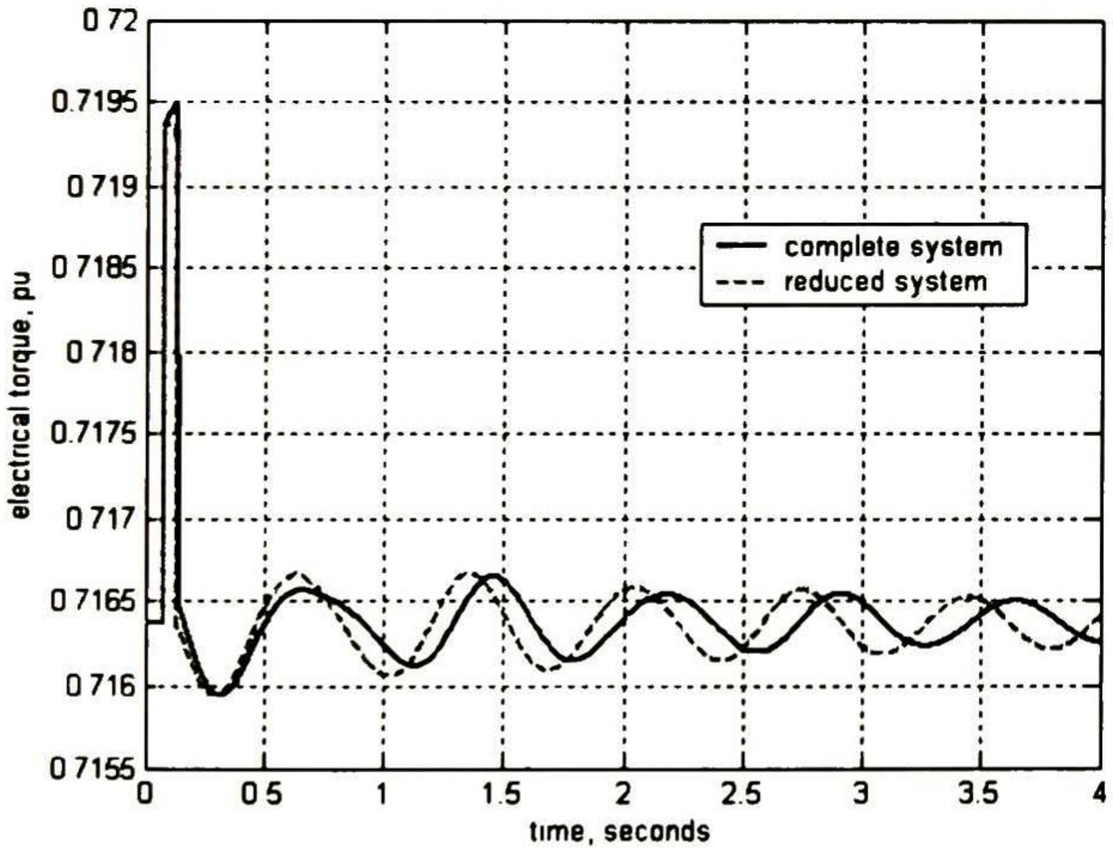



Fig. 4.42 Electrical torque from machine 1.

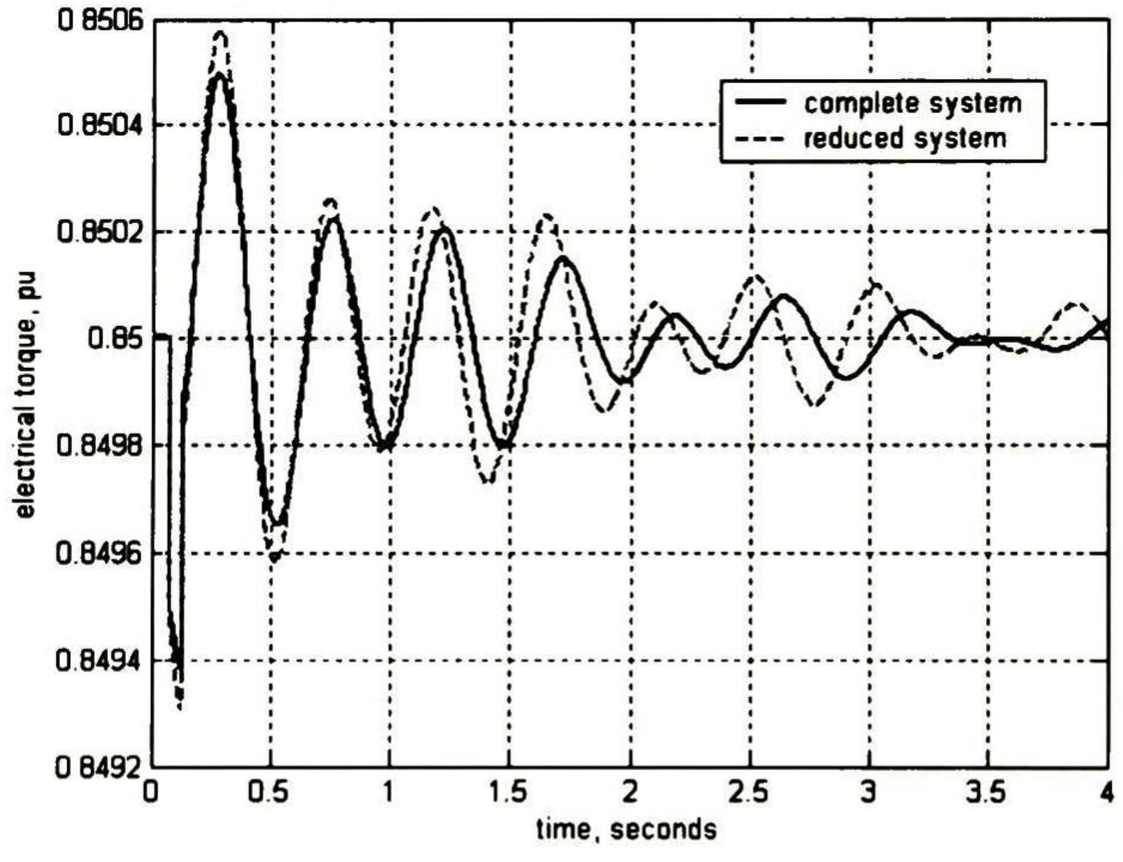



Fig. 4.43 Electrical torque from machine 3.

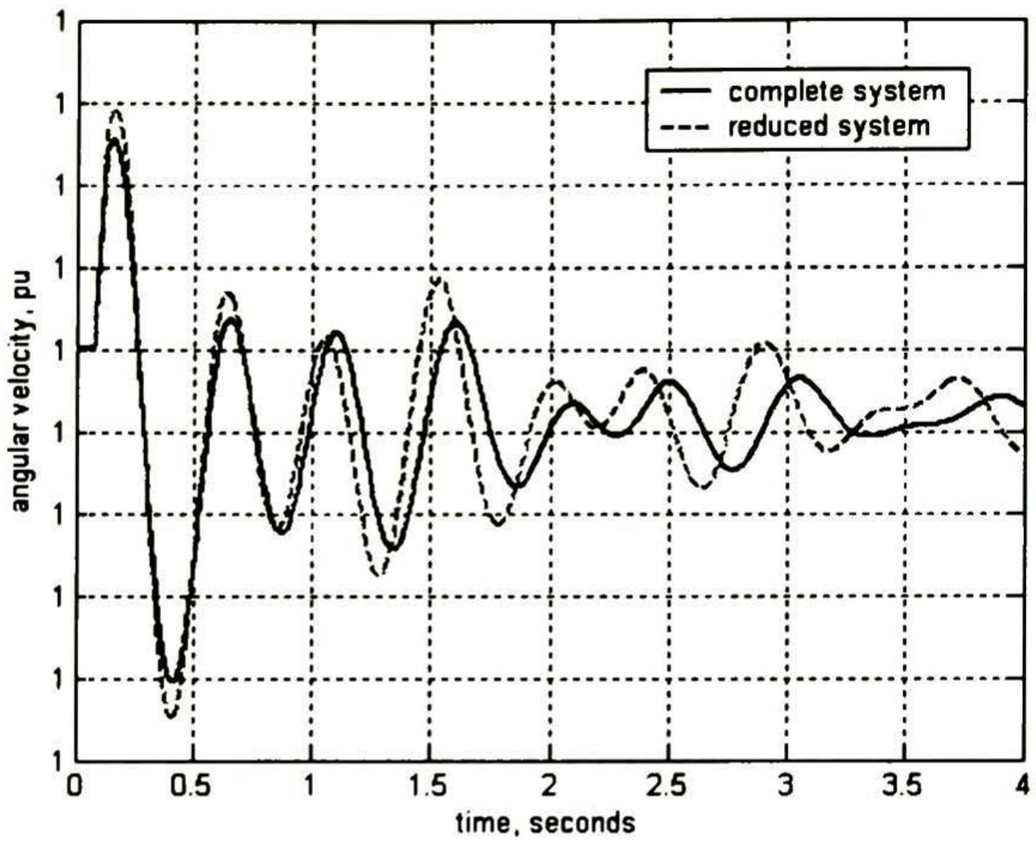



Fig. 4.44 Angular velocity behaviour from machine 3.

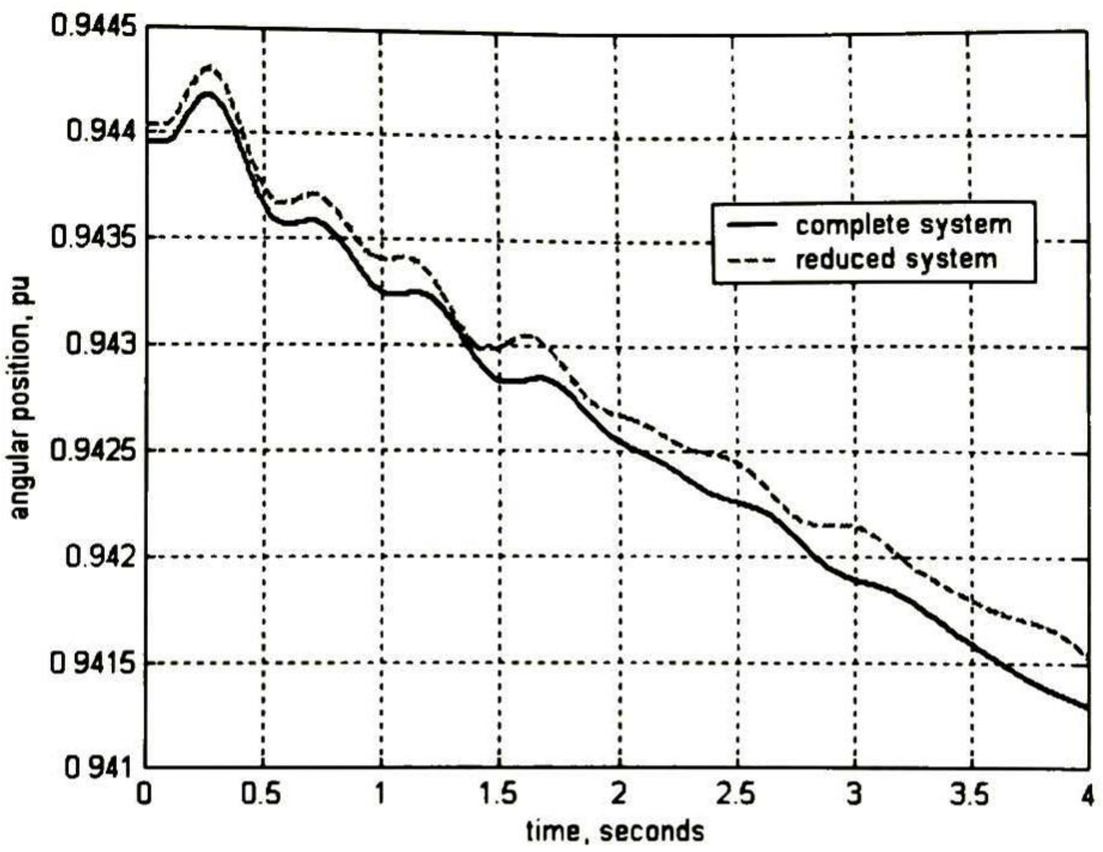



Fig. 4.45 Rotor machine's behaviour from machine 3.

4.4.2 New England multi-machine power system.

Similar to the former example, now the main objective is to foresee the complex bus voltage for the frontier nodes by an ANN, under the nominal condition [21].

The power system shown in Fig. 4.1 represents the well-know 86-buses New England multi-machine power system. The subsystem on the right of the dotted line is considered as the system under study. So, the subsystem on the left is the external system. There are two frontier nodes (1 and 9) and three frontier lines (1-2, 1-27 and 9-8). Fig. 4.46 depicts an equivalent electrical grid including two fictitious generators at nodes 1 and 9 where the complex voltage is forecasted by an ANN.

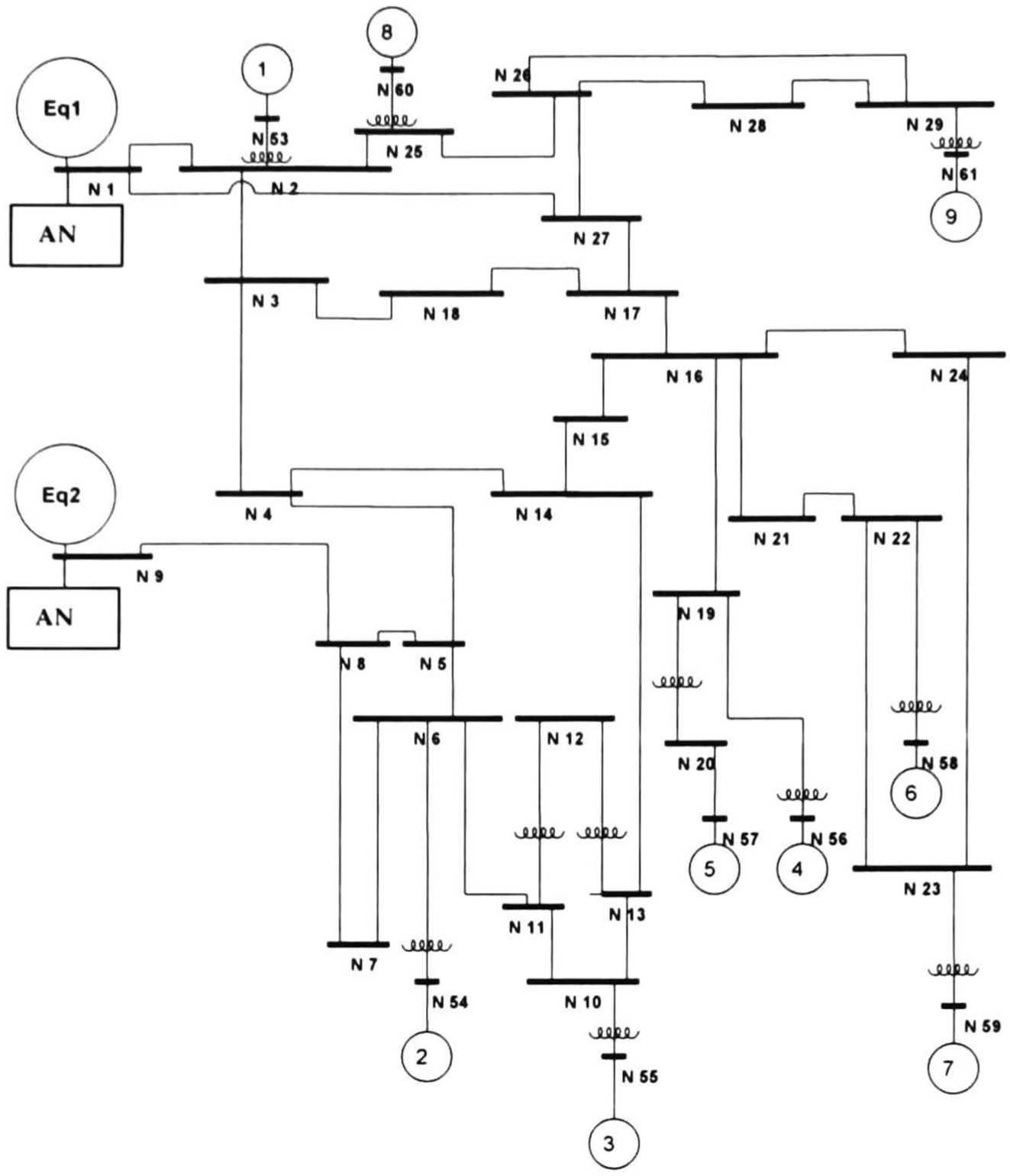



Fig. 4.46 New England 38 buses reduced multi-machine power system.

A feedforward neural network is used for predicting the complex voltage at frontier nodes. For each one is necessary to create two neural networks, one to get the real value and a second one for the imaginary part. The neural network is trained taking as input values the active and reactive (real and imaginary) power flows from several different disturbances. The input values are the active and reactive power flows from Buses 1-2 and 1-27 for the dynamic equivalent in node 1, and the corresponding flows from buses 9-8 for the dynamic equivalent on bus 9; these signals are taken from some set of disturbances applied to the complete system (Fig.4.1). The output values are considered as the complex voltage at Bus 1 and 9, respectively. The training procedure for obtaining the real and the imaginary values is done with the structure showed in Fig. 4.47 and Fig. 4.48.

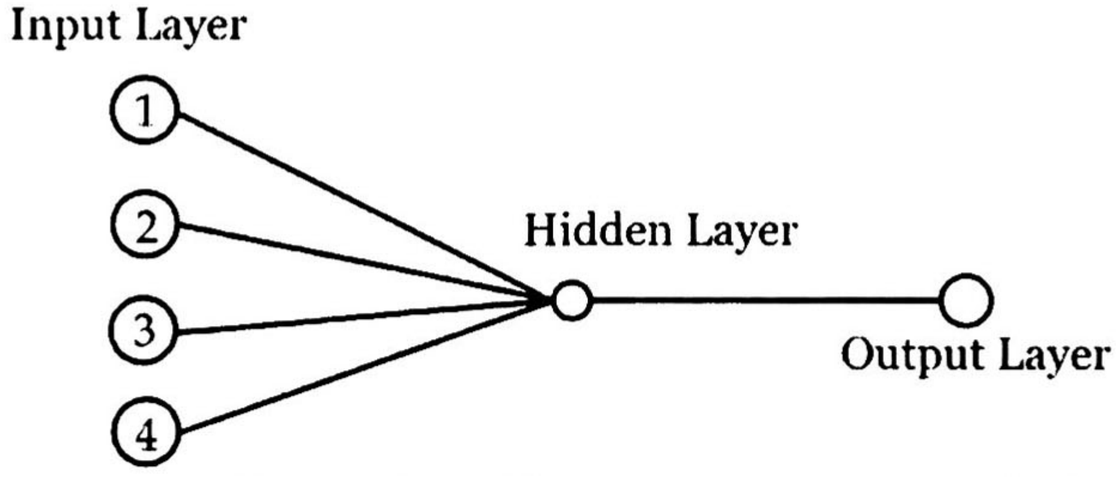



Fig. 4.47 Artificial Neural Network used for training purposes for the first equivalent.

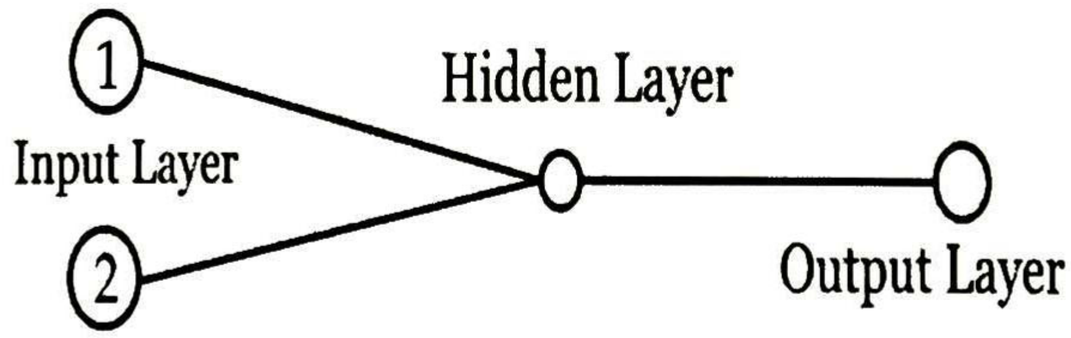



Fig. 4.48 Artificial Neural Network used for training purposes for the second equivalent.

These structures are selected after considering many other configurations and they present the best results; the hidden layer is activated by a sigmoid function (tangent-hyperbolic transfer function) while the output layer is activated by a linear function.

The ANN is trained with the Levenberg-Marquardt method, the weights from input to hidden layer and hidden to output layer are initialised randomly and the training process is done for 1000 epochs although, it always converges before the epoch number is reached.

To get the greatest trained data a previous process is done. Before training, the data set is scaled; this is done with the aim to remove the mean and scale all signals to the same variance. If the data are not scaled, the largest values tend to be dominant. For this application the variance is taken as one and the mean as zero. After the training process finishes the data need to be rescaled so, the network model can be used for any purpose.

As we already stated, the input values for training purposes are the active and reactive power flows from buses 1-2, 1-27 for the dynamic equivalent sited on bus 1, and from buses 9-8 for the dynamic equivalent sited on bus 9. These signals are obtained from different disturbances applied to the complete system, and the output values are considered as the complex voltage at Bus 1 and 9, respectively. The input values are obtained from a transient stability simulation applying several disturbances to the system in the following manner. Some disturbances consist in the variation of the parameters. Some other disturbances are variations of power load at buses 5, 7 and 9; such disturbance is performed turning up or down the power loads. The total simulation time for each disturbance is for 1s. This process is done three times for different transmission lines and load buses; it is important to remark that all the disturbances are applied to the subsystem on the right of the dotted line from Fig. 4.1. After this stage, the trained set needs to be tested or validated to put on view the training result so, it is possible to determine if the training was done in a proper manner. Figs. 4.49-4.52 depict the validation of the trained signal under other disturbances different from the ones used for training purposes.

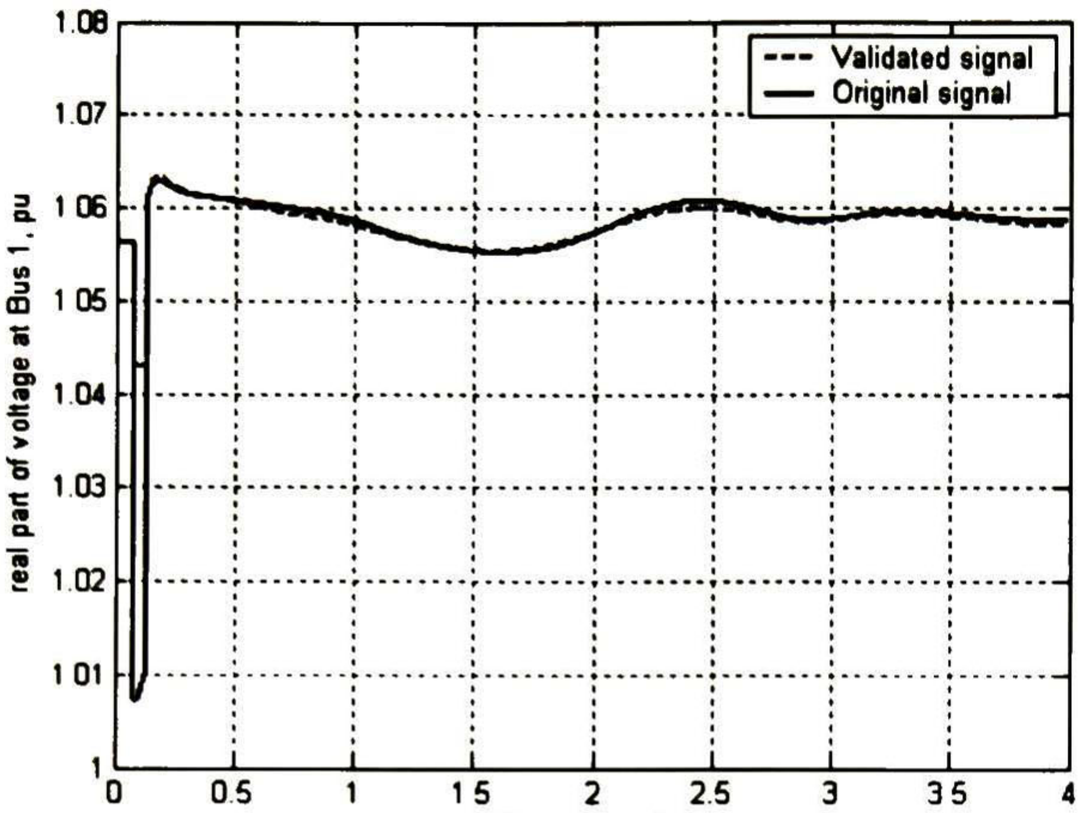



Fig. 4.49 Validation of the real part of voltage at bus 1 under a three phase fault al bus 12.

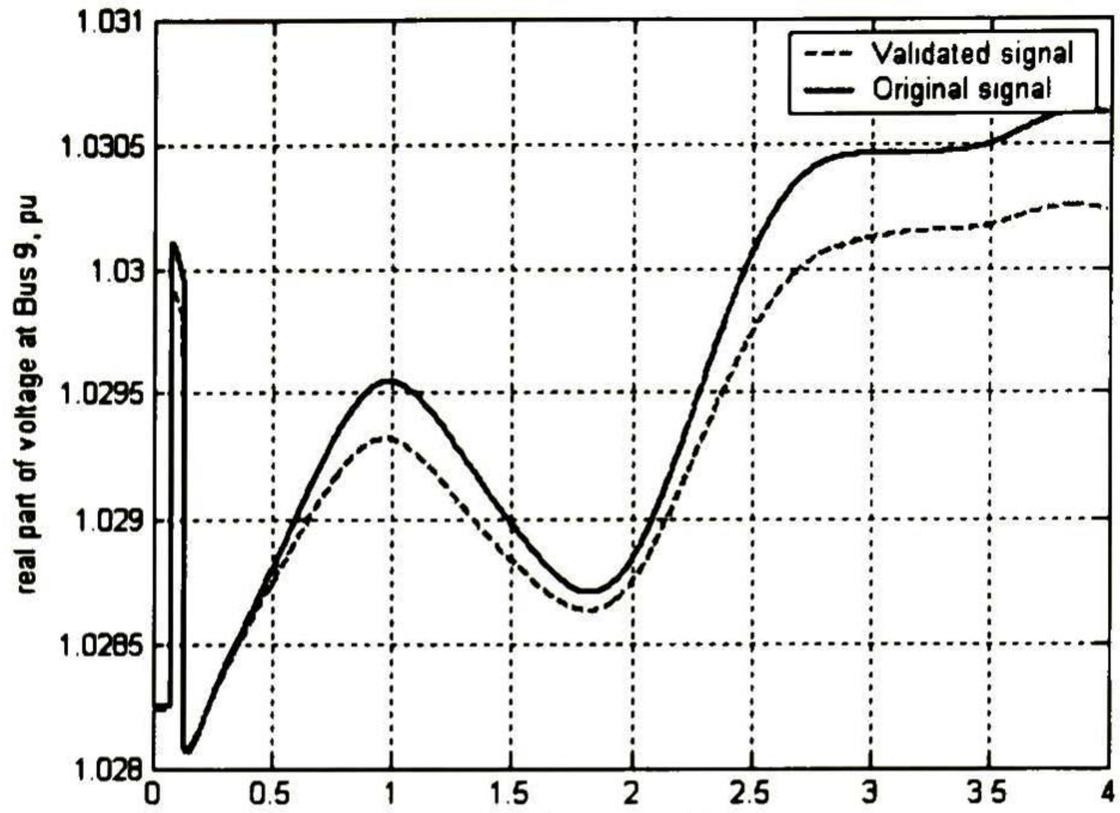



Fig. 4.50 Validation of the real part of voltage at bus 9 under a load variation at bus 3.

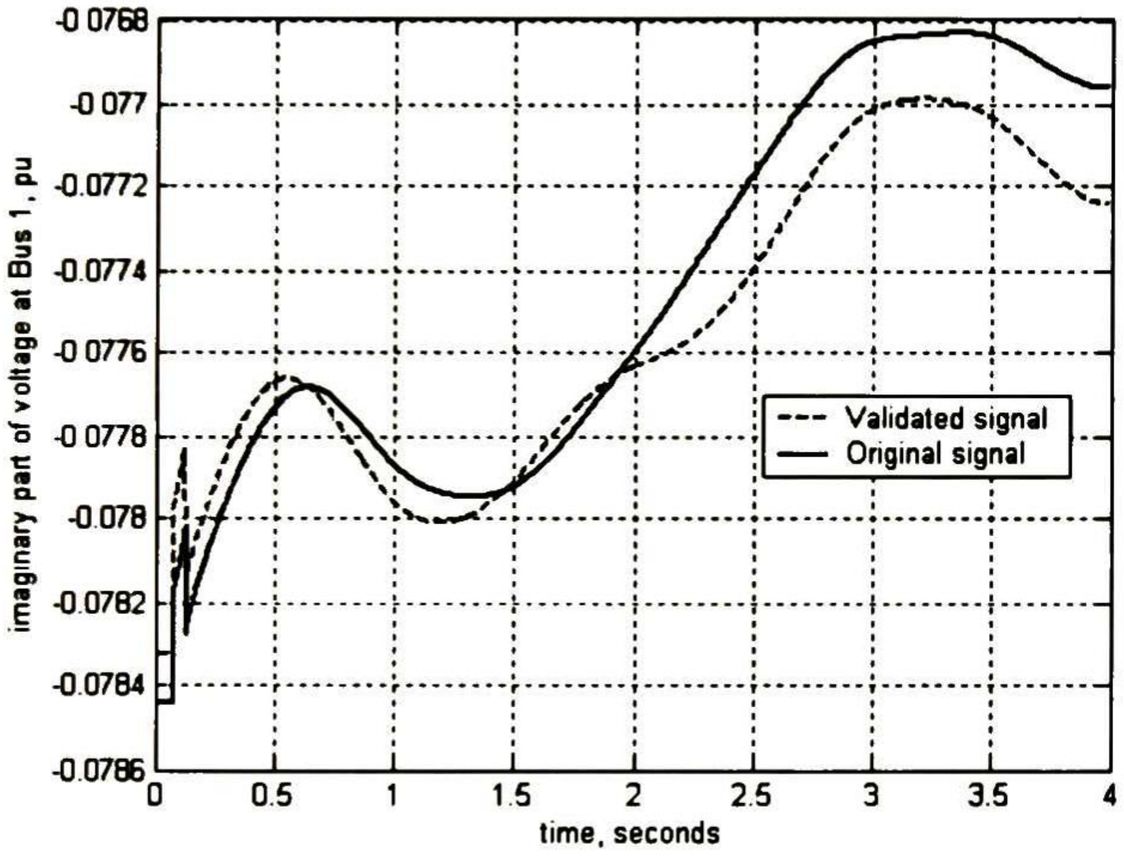



Fig. 4.51 Validation of the imaginary part of voltage at bus 1 under a variation of parameters from lines 17-27.

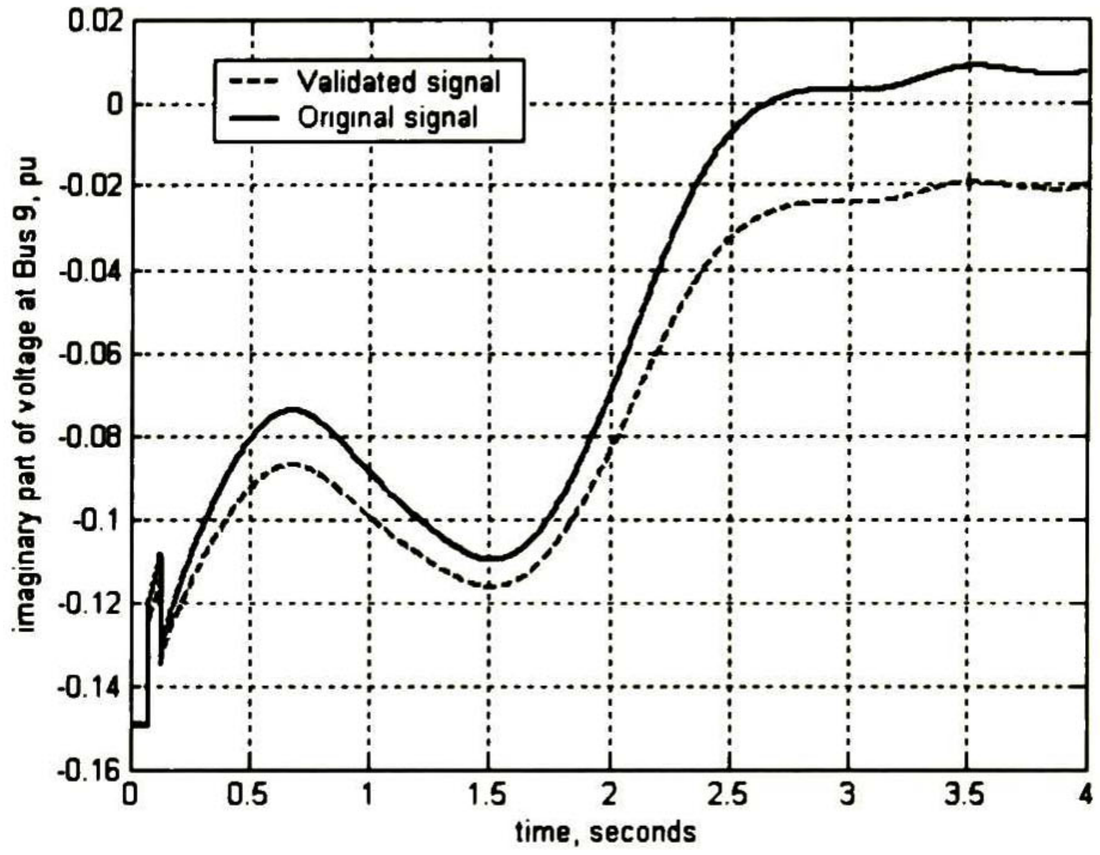



Fig. 4.52 Validation of the imaginary part of voltage at bus 19 under three phase fault at bus 16.

Once the ANNs are trained and the input signal has been validated the reduced system is tested to prove the feasibility and certainty of the dynamic equivalent.

The parameters used for the Dynamic Equivalents are taken as typical; however, the inertia constants are obtained in the following way

$$\sum_{j=1}^{N_{gen\_eq}} H_j = \sum_{\substack{i=1 \\ i \in I}}^{N_{gen\_ext}} \{H_i\} \tag{4.18}$$

$I$ is the set of generators belonging to the external system.

$N_{gen\_eq}$ is the number of equivalent generators.

$N_{gen\_ext}$ is the number of generators in the external system.

This expression is used in order to preserve the momentum.

Figs. 4.53-4.55 depict the behaviour of the reduced power system under a three phase fault at bus 4. The actual results are the simulation outputs from a transient stability programme employing the base data of the complete system (Fig. 4.1), and the predicted values are obtained after the system reduction and the substitution of the dynamic equivalents supported by an ANN that is able to reproduce the complex voltage. The dashed lines represent the predicted results.

The reduced system is tested to prove the feasibility and certainty of the dynamic equivalent in face of other disturbances such as modification of the line parameters and load variations at strategic buses. Figs. 4.56-4.60 show the performance of the reduced system under these circumstances. To make a comparison among signals, next RMS difference is computed

$$Error = \sqrt{\frac{1}{T}\int_0^T \{S_{full} - S_{equi}\}^2 dt} \tag{4.19}$$

where $S$ denote any signal. Tables 4.10-4.12 show the RMS errors encountered for each specific case.

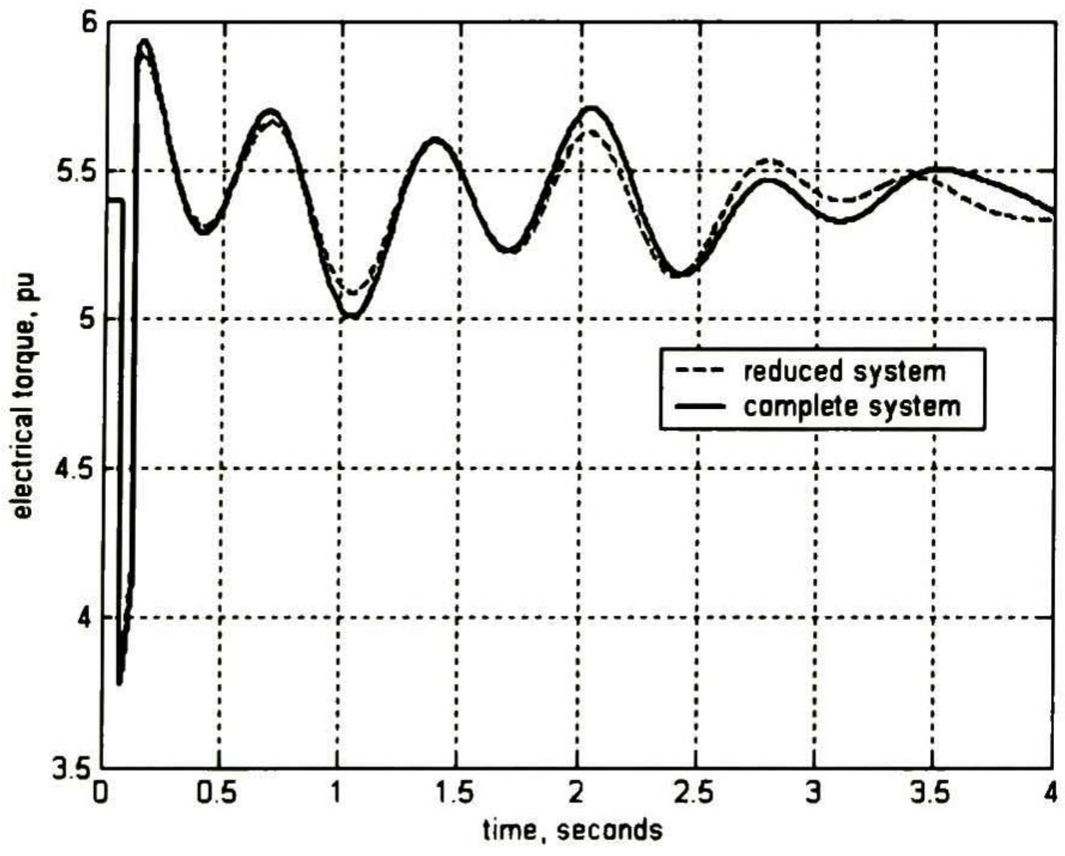



Fig. 4.53 Electrical torque from machine 8 under a 3 phase fault at bus 4.

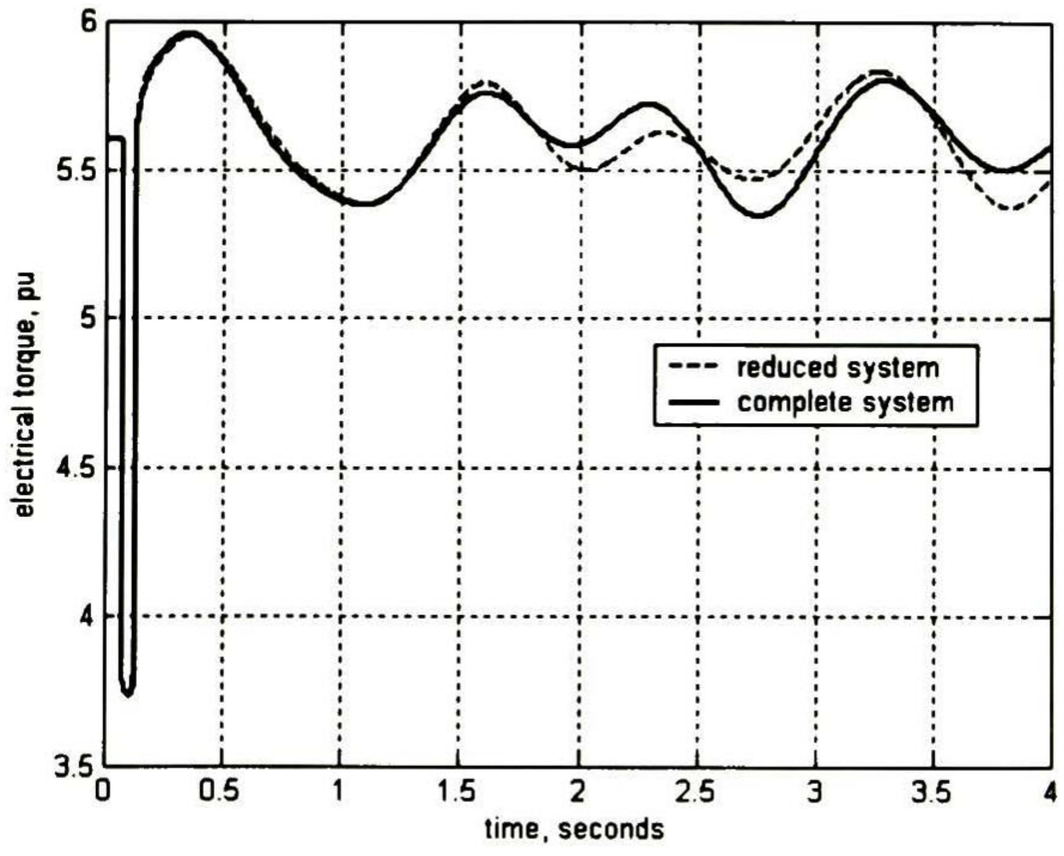



Fig. 4.54 Electrical torque from machine 7 under a 3 phase fault at bus 4.

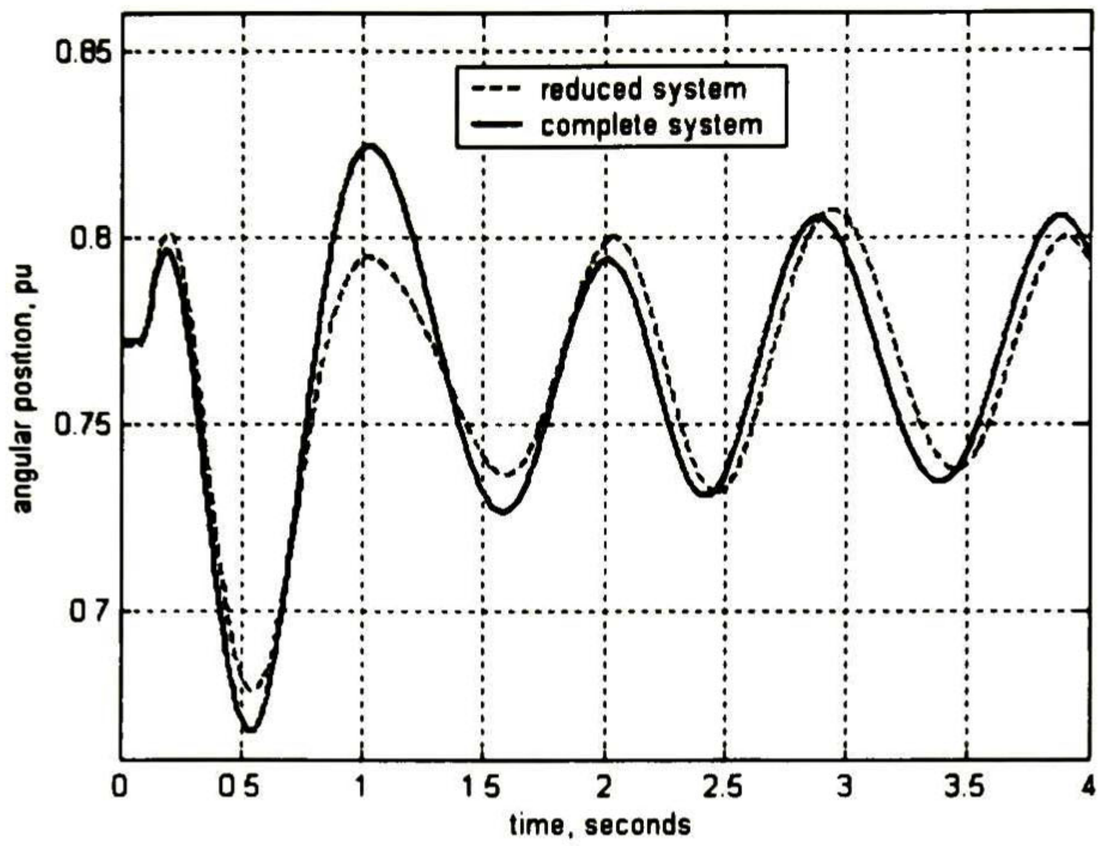



Fig. 4.55 Rotor machine's deviation machine 3 under a 3 phase fault at bus 4.

Table 4.10 RMS errors (three phase fault at Bus 4)

| Machine No. | $\delta$ (degrees) | $\omega$ (rad/s) | Pe (pu) |
|---|---|---|---|
| 1 | 2.6715 | 0.1119 | 0.0963 |
| 2 | 2.6571 | 0.1235 | 0.1012 |
| 3 | 2.7057 | 0.1245 | 0.1151 |
| 4 | 2.9934 | 0.1302 | 0.0782 |
| 5 | 3.1318 | 0.1413 | 0.0794 |
| 6 | 3.0983 | 0.1378 | 0.1027 |
| 7 | 3.0185 | 0.1322 | 0.0747 |
| 8 | 2.7537 | 0.1156 | 0.0596 |
| 9 | 2.9945 | 0.1326 | 0.1016 |

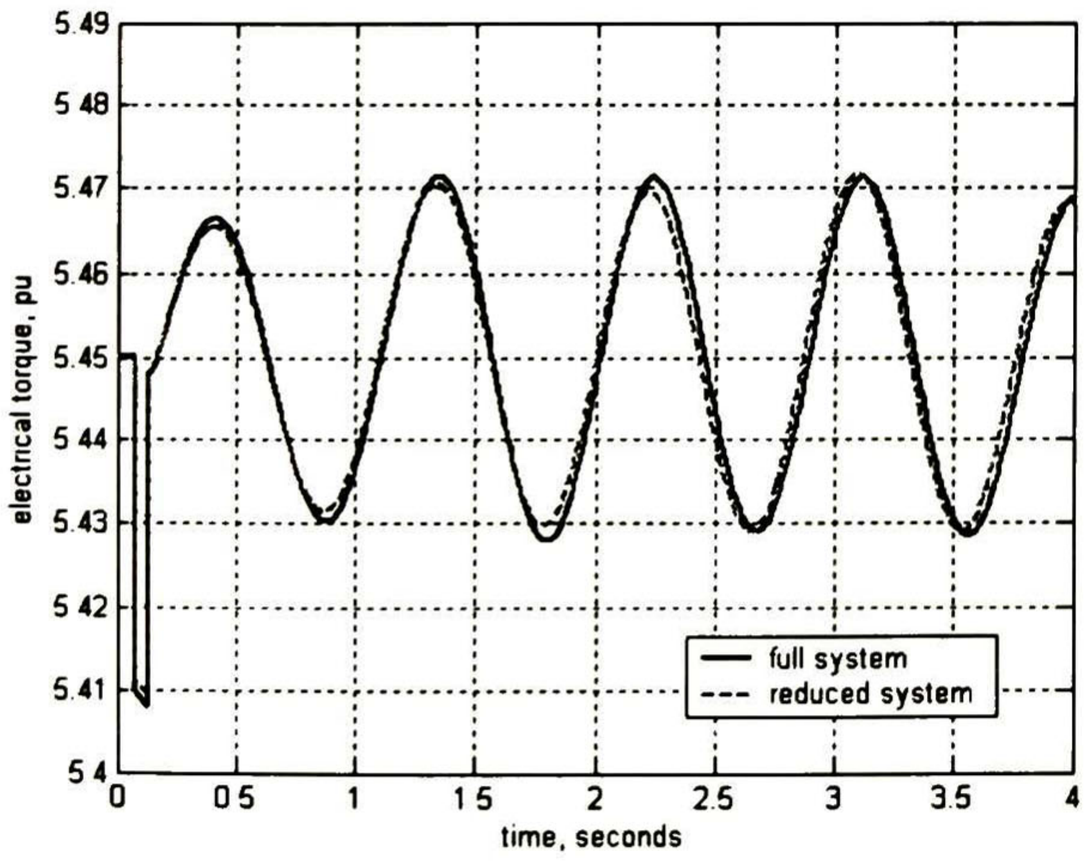



Fig. 4.56 Electrical torque from machine 2 under a variation of the parameters from line 3-18.

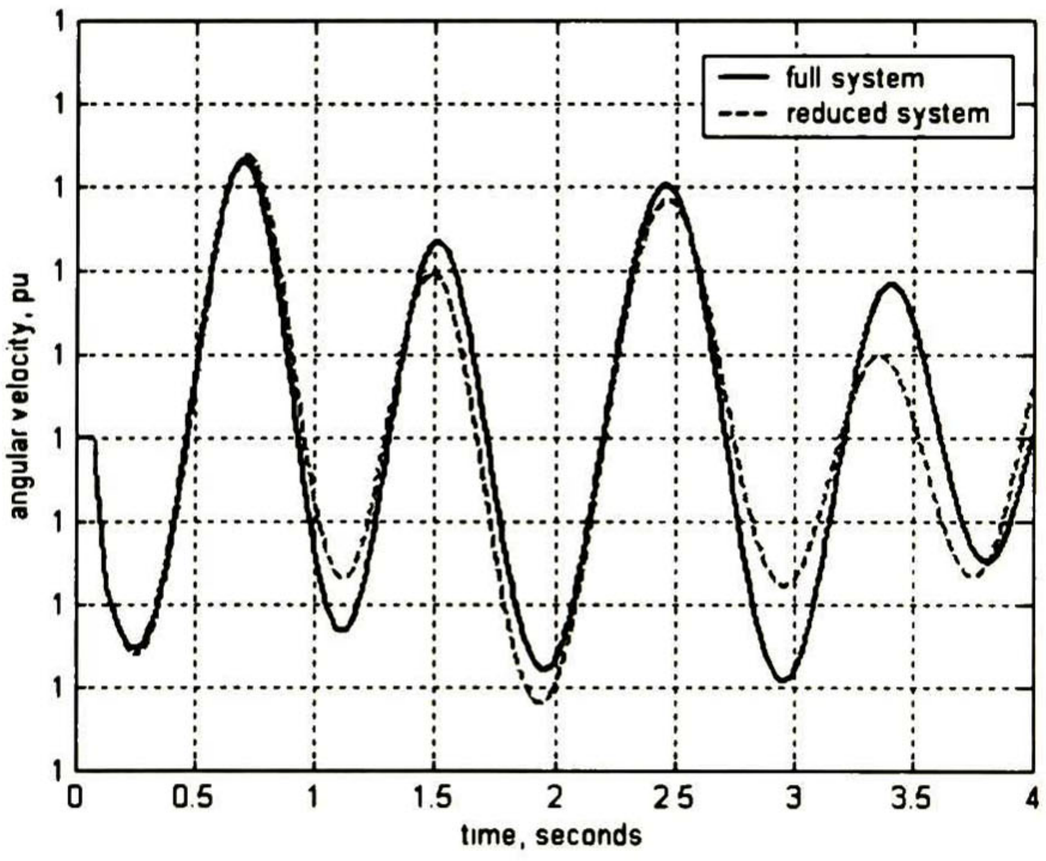



Fig. 4.57 Angular velocity from machine 5 under a variation of the parameters from line 3-18.

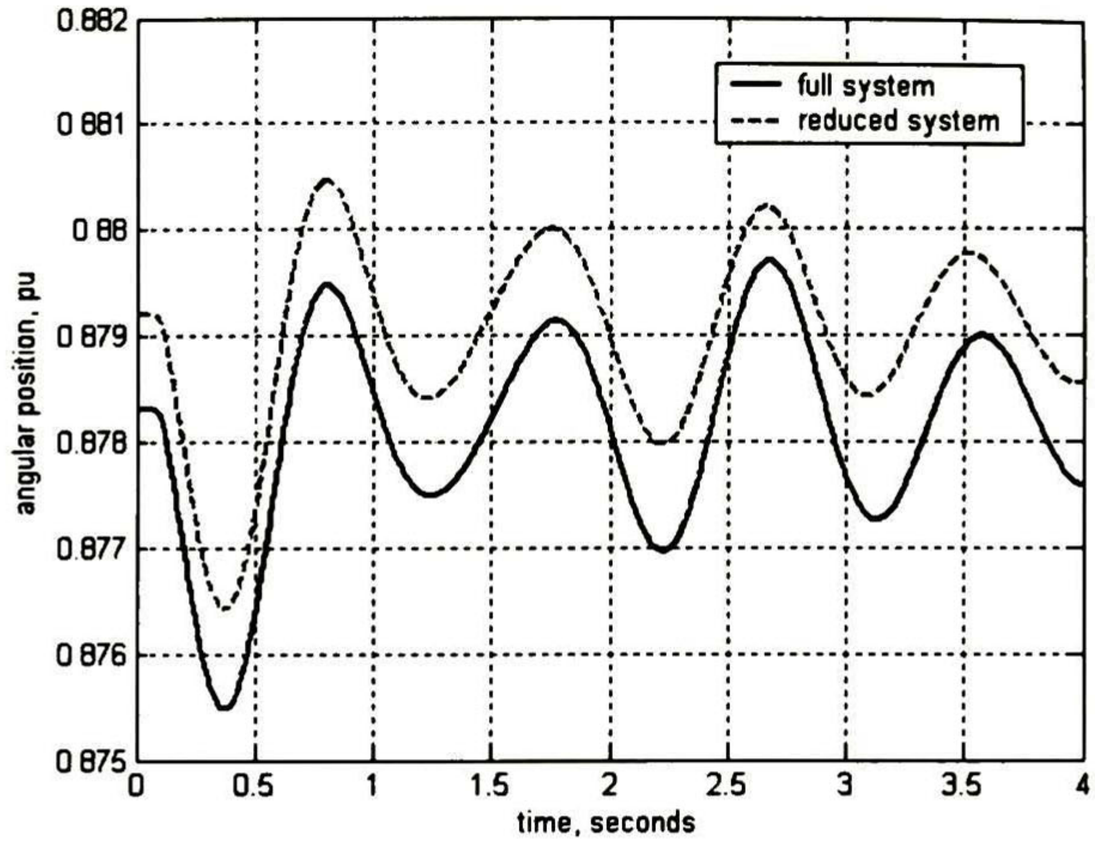



Fig. 4.58 Rotor machine's deviation from machine 4 under a variation of the parameters from line 3-18.

Table 4.11RMS errors (under a variation of the line parameters from line 3-18)

| Machine no. | $\delta$ (degrees) | $\omega$ (rad/s) | Pe (pu) |
|---|---|---|---|
| 1 | 0.1550 | 0.0047 | 0.0135 |
| 2 | 0.1913 | 0.0039 | 0.0020 |
| 3 | 0.1936 | 0.0041 | 0.0025 |
| 4 | 0.2036 | 0.0055 | 0.0030 |
| 5 | 0.2067 | 0.0059 | 0.0031 |
| 6 | 0.2063 | 0.0058 | 0.0039 |
| 7 | 0.2044 | 0.0055 | 0.0028 |
| 8 | 0.1953 | 0.0048 | 0.0022 |
| 9 | 0.2047 | 0.0059 | 0.0046 |

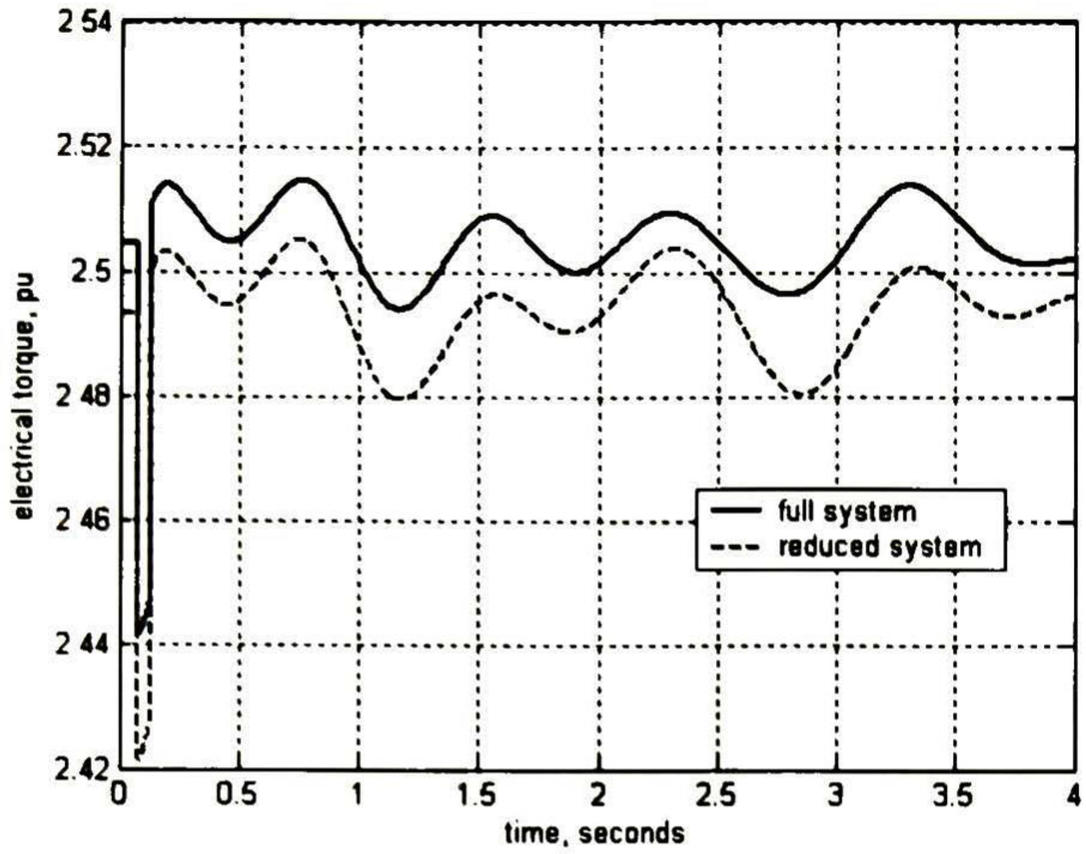



Fig. 4.59 Electrical torque from machine 1 under a load variation at bus 28.

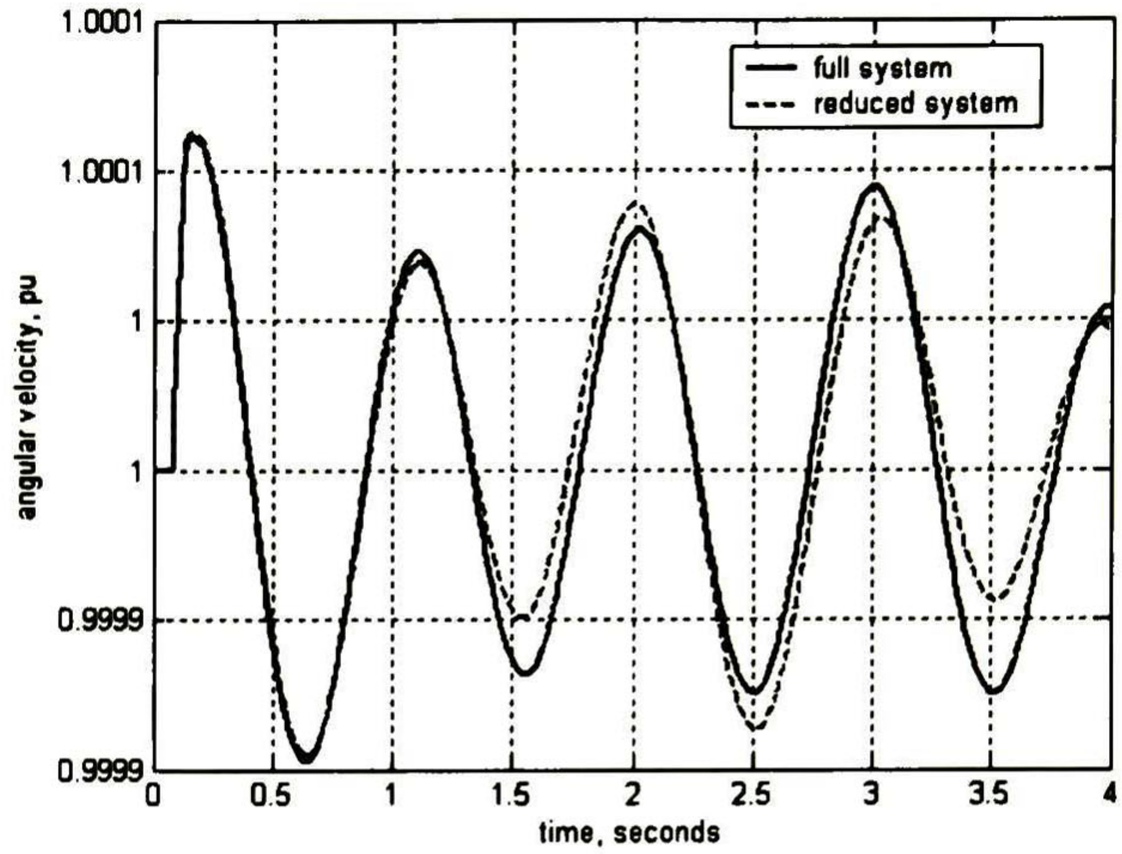



Fig. 4.60 Angular velocity from machine 9 under a load variation at bus 28.

Table 4.12 RMS errors (under a load variation at bus 28)

| Machine no. | $\delta$ (degrees) | $\omega$ (rad/s) | Pe (pu) |
|---|---|---|---|
| 1 | 0.0447 | 0.0016 | 0.0130 |
| 2 | 0.0373 | 0.0024 | 0.0024 |
| 3 | 0.0362 | 0.0021 | 0.0025 |
| 4 | 0.0370 | 0.0020 | 0.0014 |
| 5 | 0.0401 | 0.0023 | 0.0016 |
| 6 | 0.0389 | 0.0022 | 0.0020 |
| 7 | 0.0375 | 0.0020 | 0.0014 |
| 8 | 0.0308 | 0.0016 | 0.0009 |
| 9 | 0.0383 | 0.0023 | 0.0023 |

4.4.2.1 Looking for Robustness.

In order to exemplify the construction of Robust Dynamic Equivalents, three operation conditions are considered. As a consequence the states are: (1) case a, as indicated by [21]; (2) case b, Transmission lines 3-18 and 25-26, are out of service; (3) Transmission lines 4-14, 16-17 and 25-26 are out of line, case c. As well, for the three cases generators are equipped with power system stabilizers (PSS) of the type [28]

$$\frac{y(s)}{u(s)} = \frac{skT}{1+sT}\,\frac{1+sT_1}{1+sT_2}\,\frac{1+sT_3}{1+sT_4} \tag{4.20}$$

whose parameters are: $T$= 7.5, $T_1$= $T_3$=0.045, $T_2$= $T_4$=0.015, $k$=0.1

For creating Robust Dynamic Equivalents, the training route is done in the following manner. The input and output values for training purposes are the same ones like in the previous example but, in

this case the values are captured applying several disturbances to the system in the following manner. The disturbances are considered variations of parameters from lines 4-5, 13-14, 16-24, 25-26 and 23-24, and the other disturbances are obtained turning up or down the power loads at buses 3, 8, 20, 29 and 15. As soon as these signals are found the ANN is trained with all of them for the three cases mentioned above. Like in the former example the preparation of the data before and after training is done. The following course of action is completed as in the earlier example.

The reduced system, Fig. 4.46, is tested to prove the feasibility, certainty and robustness of the dynamic equivalent. Results are compared with those obtained for the full system.

The actual results are the outputs from a transient stability programme employing the base data of the complete system (Fig. 4.1); the predicted values are obtained after the system reduction and the substitution of the dynamic equivalents supported by an ANN that is able to reproduce the complex voltage. The dashed lines represent the predicted results.

The reduced system is tested under several disturbances. These disturbances are modification of the line parameters, load variations at strategic buses, and three phase faults. The disturbances applied are: i) three-phase faults at buses 12, 15 and 27; ii) line parameters variation are done at lines 19-20, 17-27 and 26-28; iii) load variation are applied at buses 4, 8 and 16. Fig. 4.61-4.63 shows the performance of the reduced system under these circumstances, case a. Fig. 4.64 displays the behaviour of the tested power system under these disturbances, case b and case c. Similar to the previous example a comparison among signals is computed. Tables 4.13-4.15 show the RMS errors encountered for all cases under each corresponding fault.

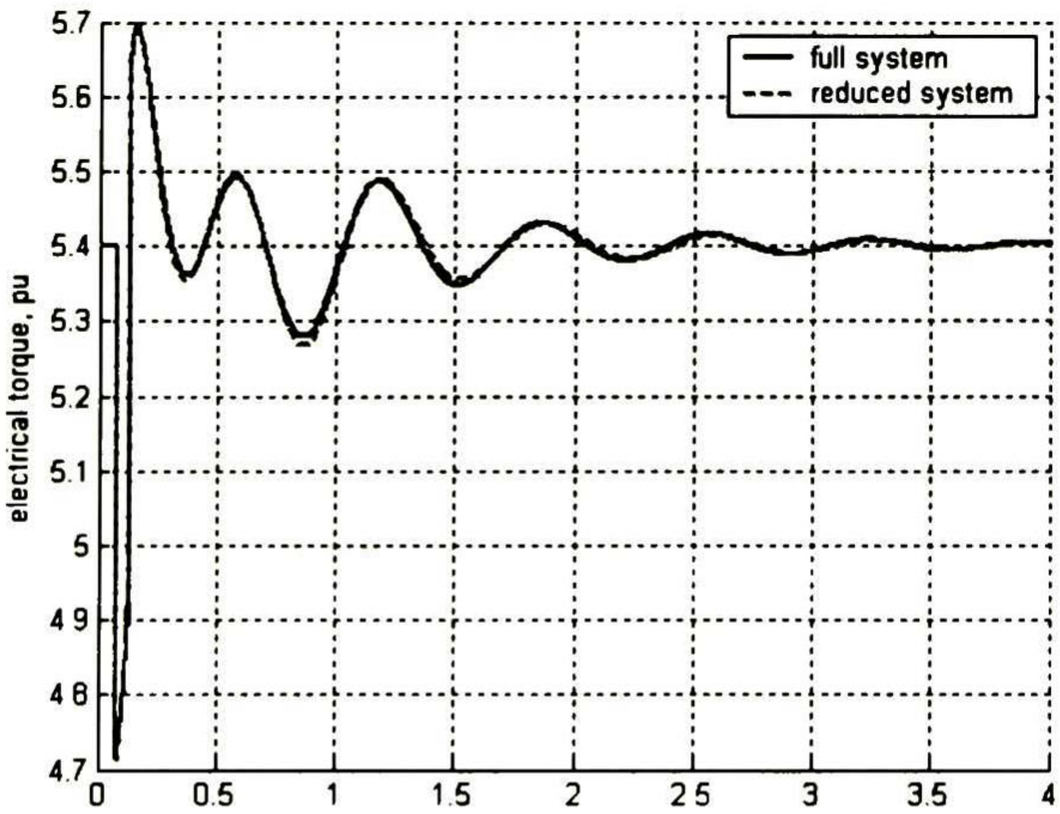



Fig. 4.61 Electrical torque from machine 8 under a 3 phase fault at bus 12, case a.

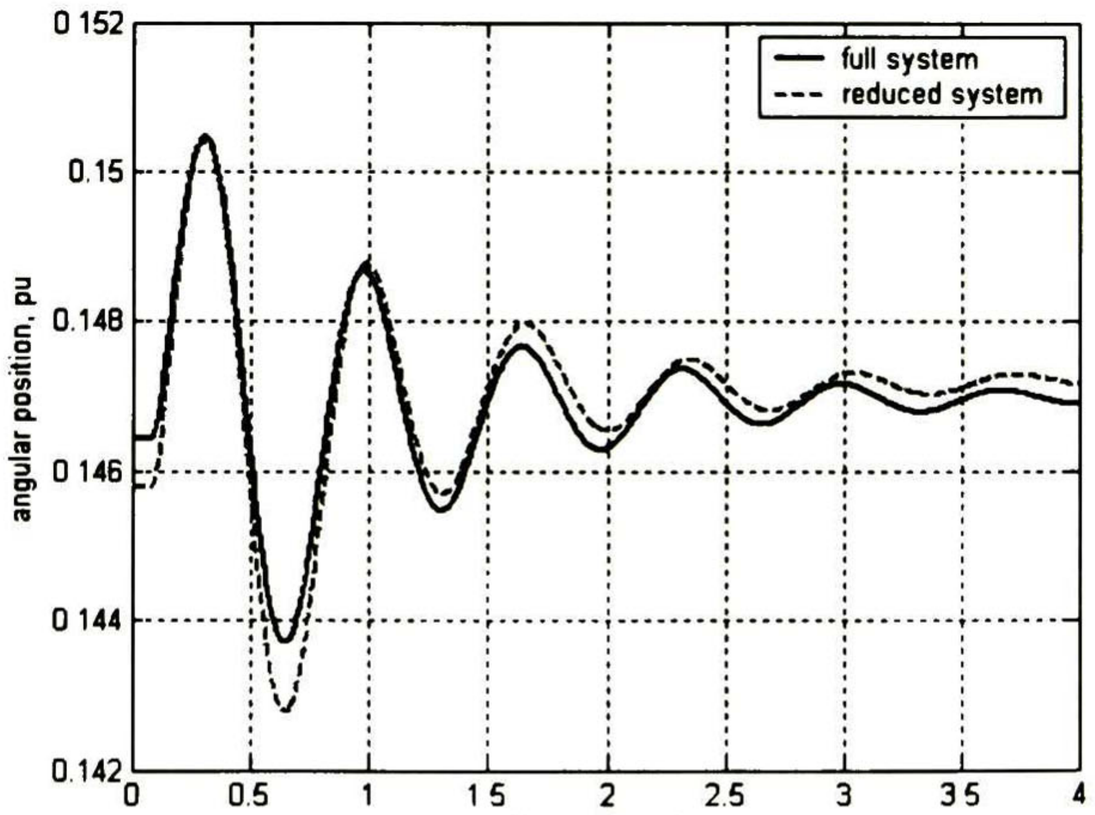



Fig. 4.62 Rotor's machine performance from machine 1 under parameters variation of line 26-28, case a.

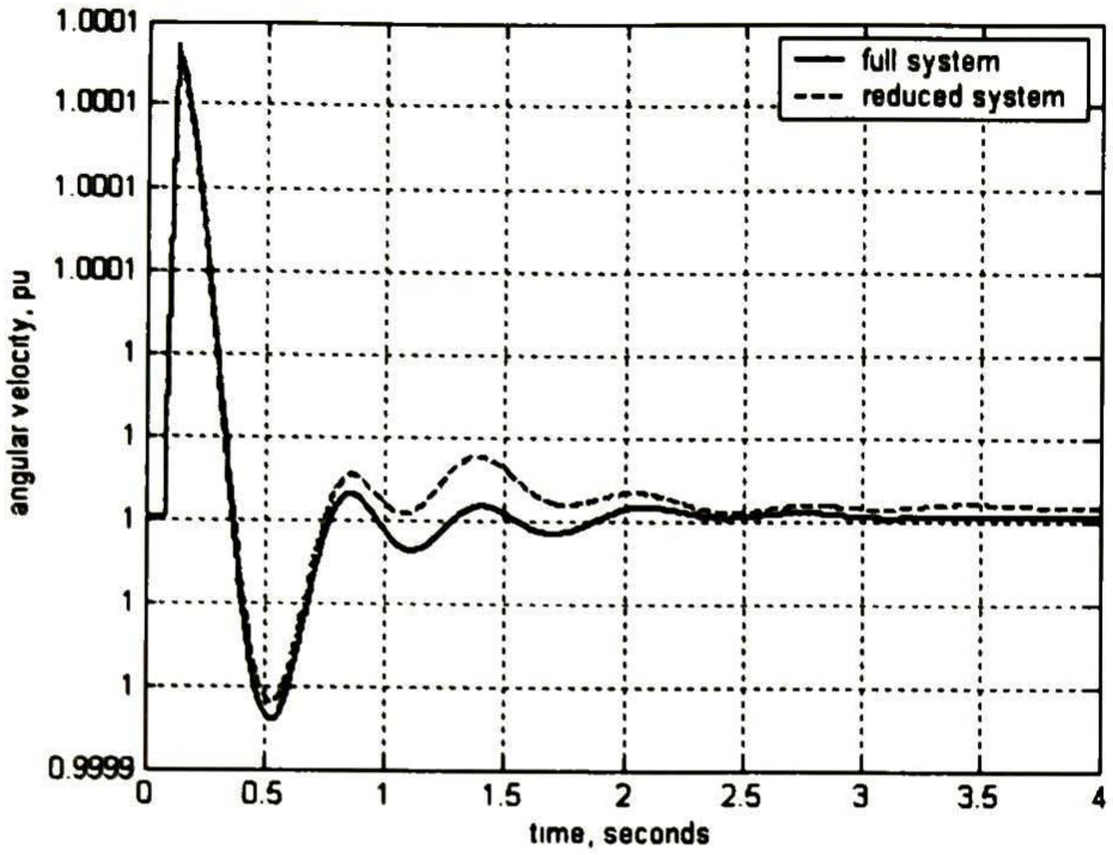



Fig. 4.63 Angular velocity from machine 6 under a load variation at bus 16, case a.

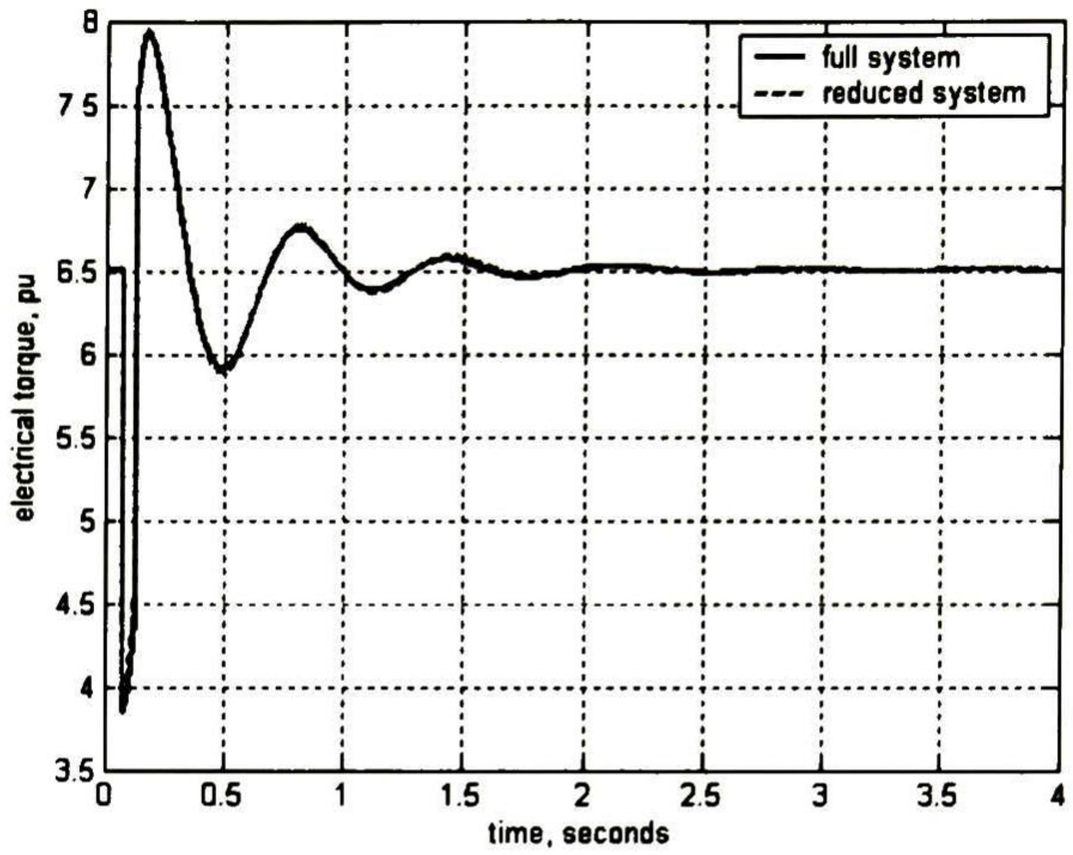



Fig. 4.64 Electrical torque from machine 3 under a 3 phase fault at bus 12, case b.

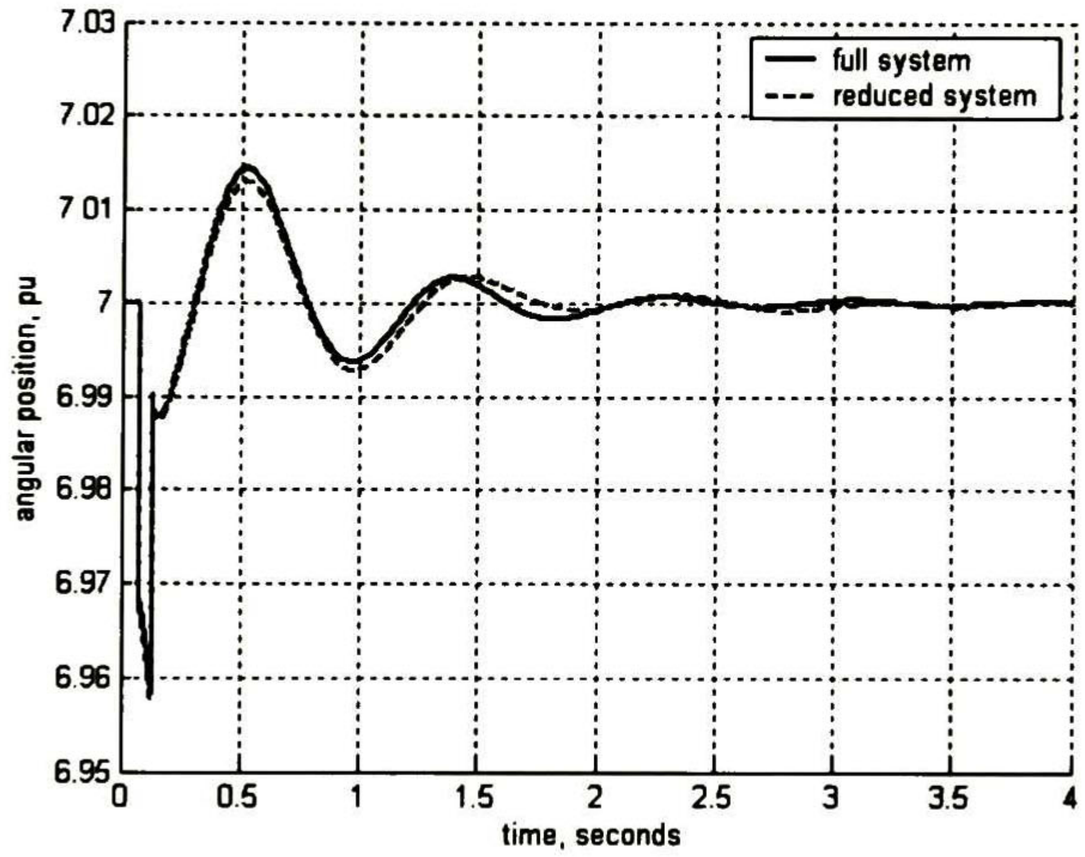



Fig. 4.65 Electrical torque from machine 5 under load variation at bus 14, case b.

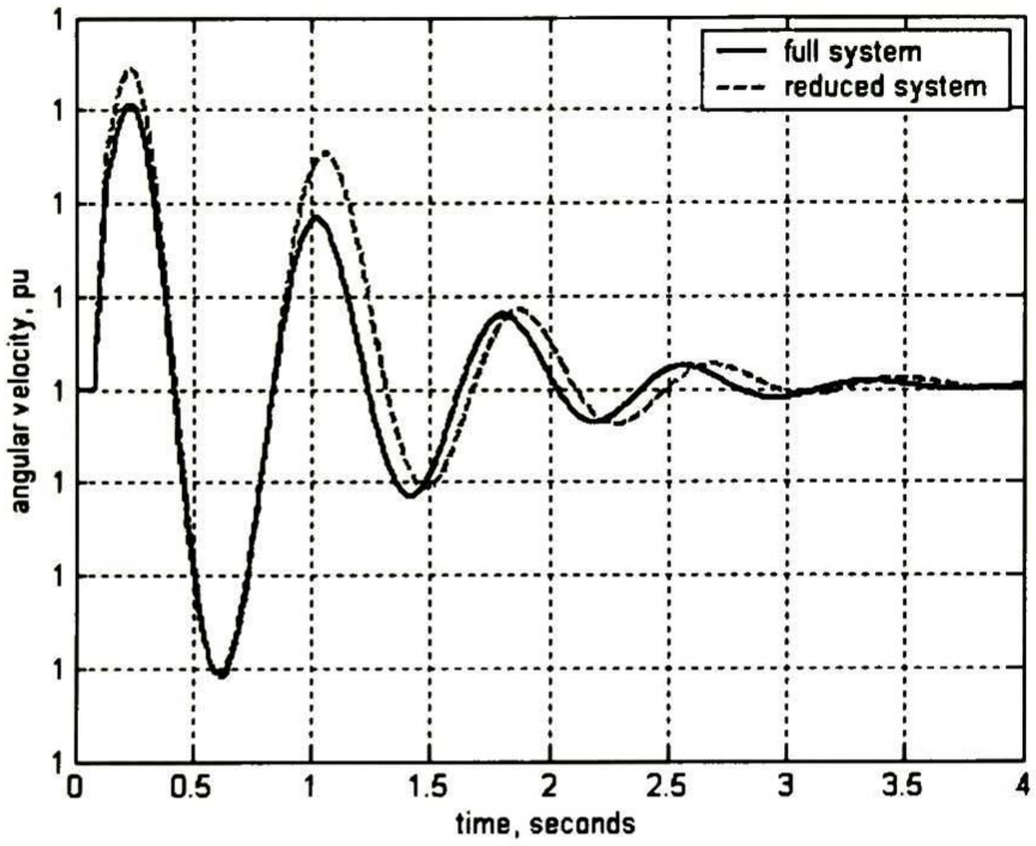



Fig. 4.66 Angular velocity from machine 1 under parameters variation of line 19-20, case b.

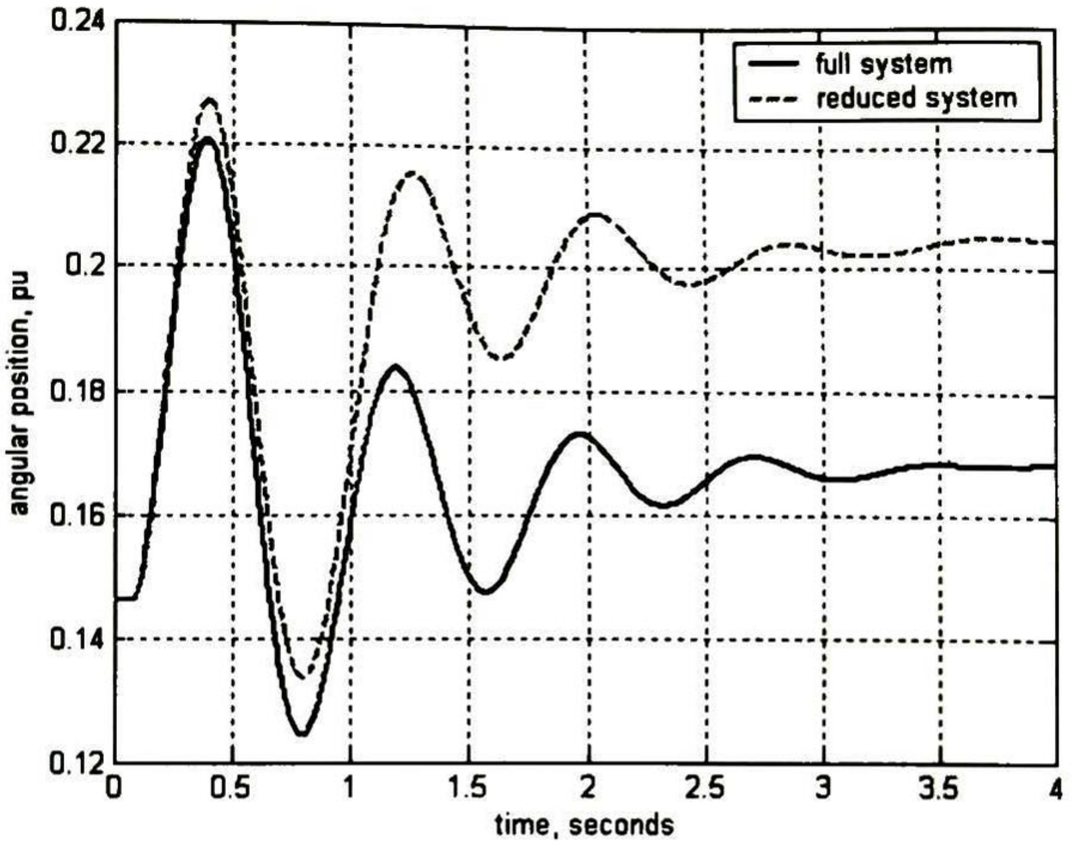



Fig. 4.67 Rotor's machine performance from machine 1 under a 3 phase fault at bus 15, case b.

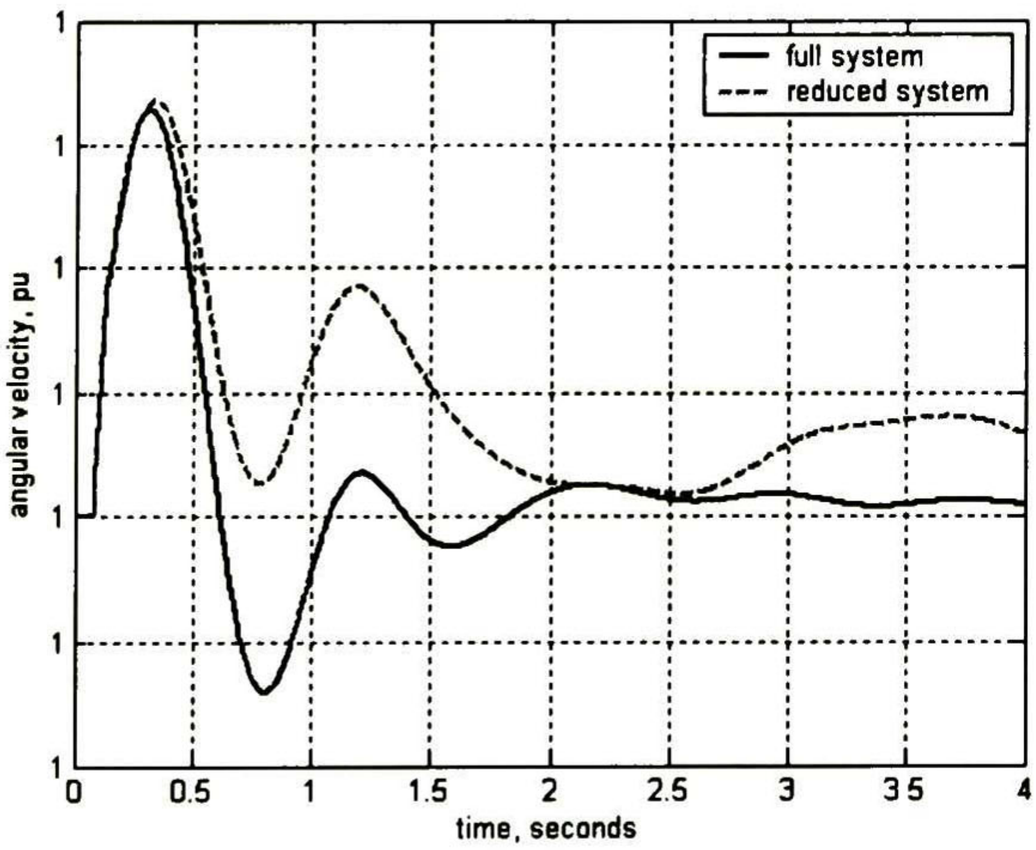



Fig. 4.68 Angular velocity from machine 7 under a load variation at bus 8, case b.

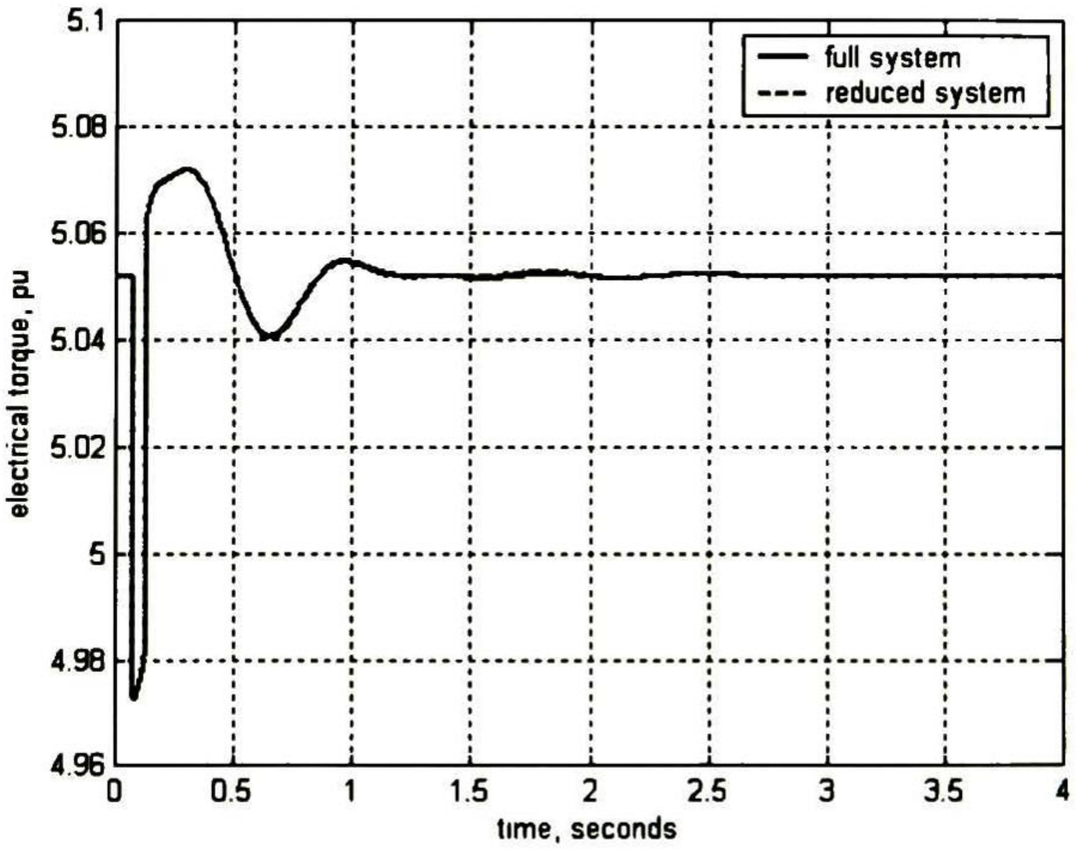



Fig. 4.69 Electrical torque from machine 5 under parameters variation of line 26-28, case b.

Table 4.13 RMS errors encountered for three phase fault conditions.

| Three-Phase fault on Bus | Machine no. | Case a | | | Case b | | | Case c | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | $\delta$ (degrees) | $\omega$ (rad/s) | Pe (pu) | $\delta$ (degrees) | $\omega$ (rad/s) | Pe (pu) | $\delta$ (degrees) | $\omega$ (rad/s) | Pe (pu) |
| 12 | 1 | 0.7856 | 0.0178 | 0.0337 | 0.8383 | 0.0281 | 0.0496 | 1.1932 | 0.0933 | 0.1653 |
| | 2 | 0.8233 | 0.0154 | 0.0217 | 0.7914 | 0.0159 | 0.0227 | 1.0688 | 0.0785 | 0.1174 |
| | 3 | 0.8187 | 0.0136 | 0.0192 | 0.7841 | 0.0138 | 0.0201 | 0.9963 | 0.0522 | 0.0899 |
| | 4 | 0.8065 | 0.0097 | 0.0063 | 0.7545 | 0.0089 | 0.0056 | 0.9130 | 0.0160 | 0.0134 |
| | 5 | 0.8046 | 0.0099 | 0.0054 | 0.7517 | 0.0091 | 0.0047 | 0.9102 | 0.0172 | 0.0138 |
| | 6 | 0.8118 | 0.0105 | 0.0087 | 0.7592 | 0.0095 | 0.0074 | 0.9209 | 0.0202 | 0.0240 |
| | 7 | 0.8108 | 0.0100 | 0.0061 | 0.7580 | 0.0092 | 0.0053 | 0.9168 | 0.0175 | 0.0144 |
| | 8 | 0.8157 | 0.0104 | 0.0062 | 0.8144 | 0.0143 | 0.0118 | 1.0335 | 0.0485 | 0.0442 |
| | 9 | 0.8112 | 0.0096 | 0.0064 | 0.7533 | 0.0086 | 0.0049 | 0.9829 | 0.0325 | 0.0378 |
| 15 | 1 | 1.8533 | 0.0329 | 0.0524 | 2.0533 | 0.0391 | 0.0589 | 2.1003 | 0.1165 | 0.1864 |
| | 2 | 1.8655 | 0.0274 | 0.0282 | 1.9520 | 0.0270 | 0.0254 | 1.9349 | 0.0920 | 0.1265 |
| | 3 | 1.8631 | 0.0248 | 0.0259 | 1.9460 | 0.0247 | 0.0239 | 1.8769 | 0.0645 | 0.0988 |
| | 4 | 1.8400 | 0.0199 | 0.0088 | 1.8823 | 0.0190 | 0.0068 | 1.7776 | 0.0281 | 0.0201 |
| | 5 | 1.8355 | 0.0202 | 0.0085 | 1.8760 | 0.0192 | 0.0066 | 1.7723 | 0.0298 | 0.0206 |
| | 6 | 1.8507 | 0.0212 | 0.0138 | 1.8928 | 0.0197 | 0.0097 | 1.7902 | 0.0338 | 0.0347 |
| | 7 | 1.8489 | 0.0204 | 0.0091 | 1.8908 | 0.0193 | 0.0067 | 1.7846 | 0.0302 | 0.0214 |
| | 8 | 1.8666 | 0.0238 | 0.0202 | 2.0197 | 0.0269 | 0.0208 | 1.9379 | 0.0699 | 0.0570 |
| | 9 | 1.8534 | 0.0212 | 0.0192 | 1.8803 | 0.0189 | 0.0097 | 1.8875 | 0.0514 | 0.0536 |
| 27 | 1 | 1.3782 | 0.0298 | 0.0514 | 1.5935 | 0.0259 | 0.0366 | 1.6839 | 0.1370 | 0.2329 |
| | 2 | 1.4052 | 0.0225 | 0.0219 | 1.5222 | 0.0204 | 0.0144 | 1.3289 | 0.0609 | 0.0751 |
| | 3 | 1.4019 | 0.0204 | 0.0196 | 1.5174 | 0.0189 | 0.0136 | 1.2823 | 0.0451 | 0.0609 |
| | 4 | 1.3819 | 0.0167 | 0.0088 | 1.4675 | 0.0151 | 0.0052 | 1.2059 | 0.0237 | 0.0175 |
| | 5 | 1.3759 | 0.0169 | 0.0081 | 1.4614 | 0.0152 | 0.0050 | 1.2041 | 0.0250 | 0.0174 |
| | 6 | 1.3893 | 0.0180 | 0.0136 | 1.4750 | 0.0156 | 0.0072 | 1.2181 | 0.0281 | 0.0275 |
| | 7 | 1.3875 | 0.0172 | 0.0091 | 1.4734 | 0.0153 | 0.0051 | 1.2123 | 0.0254 | 0.0180 |
| | 8 | 1.4108 | 0.0224 | 0.0293 | 1.5774 | 0.0210 | 0.0213 | 1.4445 | 0.0790 | 0.0740 |
| | 9 | 1.3936 | 0.0187 | 0.0226 | 1.4662 | 0.0150 | 0.0078 | 1.3547 | 0.0565 | 0.0681 |

As it is noticed from the RMS errors depicted in Table 4.13, the feasibility and robustness from the Dynamic Equivalent is corroborated. Even though the errors obtained for the rotor's machine performance (angular deviation) which are 1 or 2 degrees for some cases, it is not a bad indicator because the errors encountered for the electrical power and for the speed deviation are really minor.

Table 4.14 RMS errors encountered for line variation faults.

| Line From-To | Machine no. | Case a | | | Case b | | | Case c | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | $\delta$ (degrees) | $\omega$ (rad/s) | Pe (pu) | $\delta$ (degrees) | $\omega$ (rad/s) | Pe (pu) | $\delta$ (degrees) | $\omega$ (rad/s) | Pe (pu) |
| 16-24 | 1 | 0.0320 | 0.0006 | 0.0129 | 0.0308 | 0.0015 | 0.0027 | 0.0652 | 0.0060 | 0.0098 |
| | 2 | 0.0286 | 0.0002 | 0.0002 | 0.0256 | 0.0008 | 0.0008 | 0.0508 | 0.0042 | 0.0056 |
| | 3 | 0.0288 | 0.0002 | 0.0002 | 0.0254 | 0.0007 | 0.0007 | 0.0457 | 0.0029 | 0.0044 |
| | 4 | 0.0273 | 0.0001 | 0.0001 | 0.0235 | 0.0005 | 0.0002 | 0.0384 | 0.0011 | 0.0009 |
| | 5 | 0.0277 | 0.0002 | 0.0001 | 0.0235 | 0.0005 | 0.0002 | 0.0383 | 0.0012 | 0.0009 |
| | 6 | 0.0275 | 0.0002 | 0.0002 | 0.0239 | 0.0005 | 0.0003 | 0.0394 | 0.0014 | 0.0015 |
| | 7 | 0.0277 | 0.0002 | 0.0001 | 0.0238 | 0.0005 | 0.0002 | 0.0388 | 0.0012 | 0.0009 |
| | 8 | 0.0224 | 0.0003 | 0.0004 | 0.0285 | 0.0009 | 0.0010 | 0.0509 | 0.0035 | 0.0031 |
| | 9 | 0.0255 | 0.0002 | 0.0003 | 0.0237 | 0.0005 | 0.0003 | 0.0464 | 0.0024 | 0.0027 |
| 19-20 | 1 | 0.0547 | 0.0018 | 0.0134 | 0.0094 | 0.0001 | 0.0002 | 0.0165 | 0.0007 | 0.0011 |
| | 2 | 0.0245 | 0.0009 | 0.0010 | 0.0112 | 0.0001 | 0.0001 | 0.0139 | 0.0004 | 0.0004 |
| | 3 | 0.0242 | 0.0007 | 0.0009 | 0.0107 | 0.0001 | 0.0001 | 0.0140 | 0.0003 | 0.0004 |
| | 4 | 0.0228 | 0.0005 | 0.0005 | 0.0095 | 0.0001 | 0.0000 | 0.0138 | 0.0002 | 0.0001 |
| | 5 | 0.0231 | 0.0006 | 0.0004 | 0.0093 | 0.0001 | 0.0000 | 0.0138 | 0.0002 | 0.0001 |
| | 6 | 0.0232 | 0.0006 | 0.0006 | 0.0095 | 0.0001 | 0.0000 | 0.0139 | 0.0002 | 0.0002 |
| | 7 | 0.0232 | 0.0006 | 0.0004 | 0.0094 | 0.0001 | 0.0000 | 0.0139 | 0.0002 | 0.0001 |
| | 8 | 0.0221 | 0.0010 | 0.0014 | 0.0107 | 0.0001 | 0.0001 | 0.0169 | 0.0005 | 0.0004 |
| | 9 | 0.0226 | 0.0007 | 0.0009 | 0.0094 | 0.0001 | 0.0000 | 0.0154 | 0.0004 | 0.0004 |
| 28-29 | 1 | 0.0224 | 0.0018 | 0.0135 | 0.0671 | 0.0012 | 0.0022 | 0.1004 | 0.0084 | 0.0142 |
| | 2 | 0.0653 | 0.0007 | 0.0007 | 0.0665 | 0.0008 | 0.0005 | 0.0752 | 0.0035 | 0.0040 |
| | 3 | 0.0654 | 0.0007 | 0.0005 | 0.0658 | 0.0008 | 0.0005 | 0.0728 | 0.0026 | 0.0034 |
| | 4 | 0.0633 | 0.0006 | 0.0006 | 0.0628 | 0.0006 | 0.0002 | 0.0685 | 0.0014 | 0.0011 |
| | 5 | 0.0635 | 0.0006 | 0.0004 | 0.0625 | 0.0006 | 0.0002 | 0.0685 | 0.0015 | 0.0011 |
| | 6 | 0.0637 | 0.0006 | 0.0007 | 0.0631 | 0.0006 | 0.0003 | 0.0693 | 0.0017 | 0.0017 |
| | 7 | 0.0638 | 0.0006 | 0.0005 | 0.0630 | 0.0006 | 0.0002 | 0.0689 | 0.0015 | 0.0011 |
| | 8 | 0.0594 | 0.0010 | 0.0017 | 0.0674 | 0.0009 | 0.0011 | 0.0849 | 0.0049 | 0.0045 |
| | 9 | 0.0617 | 0.0007 | 0.0014 | 0.0626 | 0.0006 | 0.0004 | 0.0782 | 0.0034 | 0.0040 |

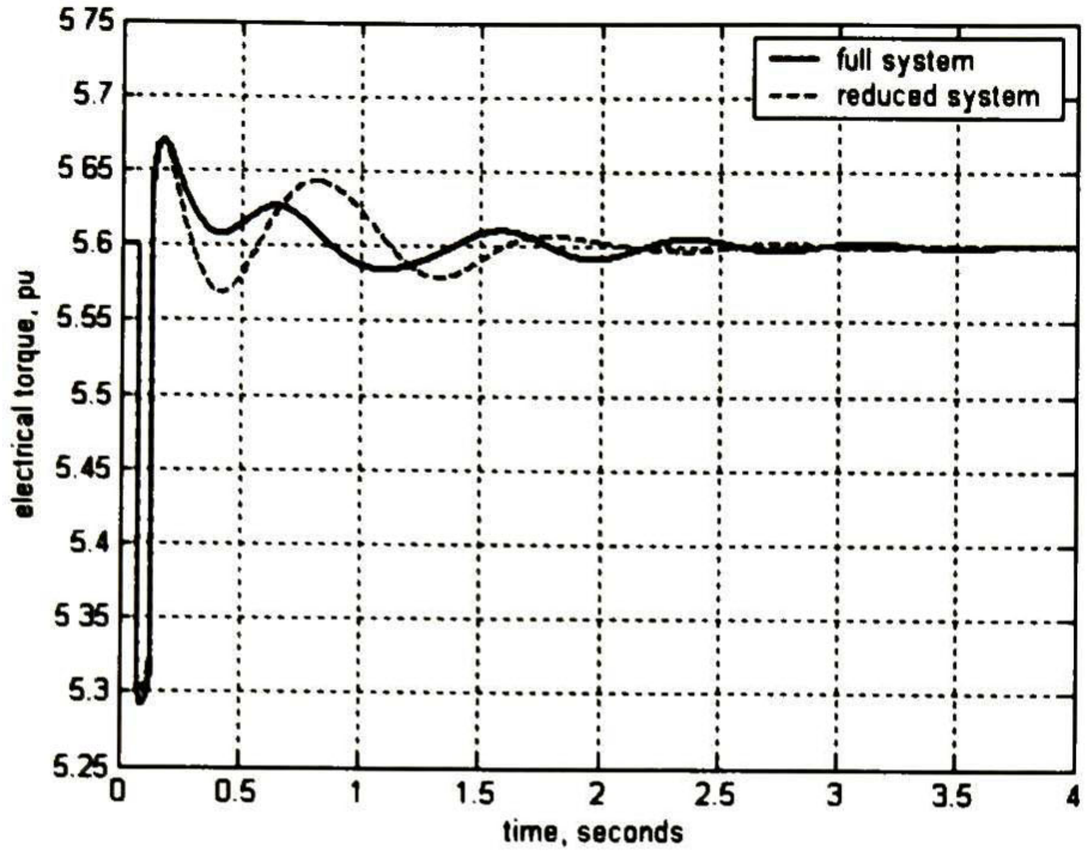



Fig. 4.70 Electrical torque from machine 7 under a 3 phase fault at bus 27, case c.

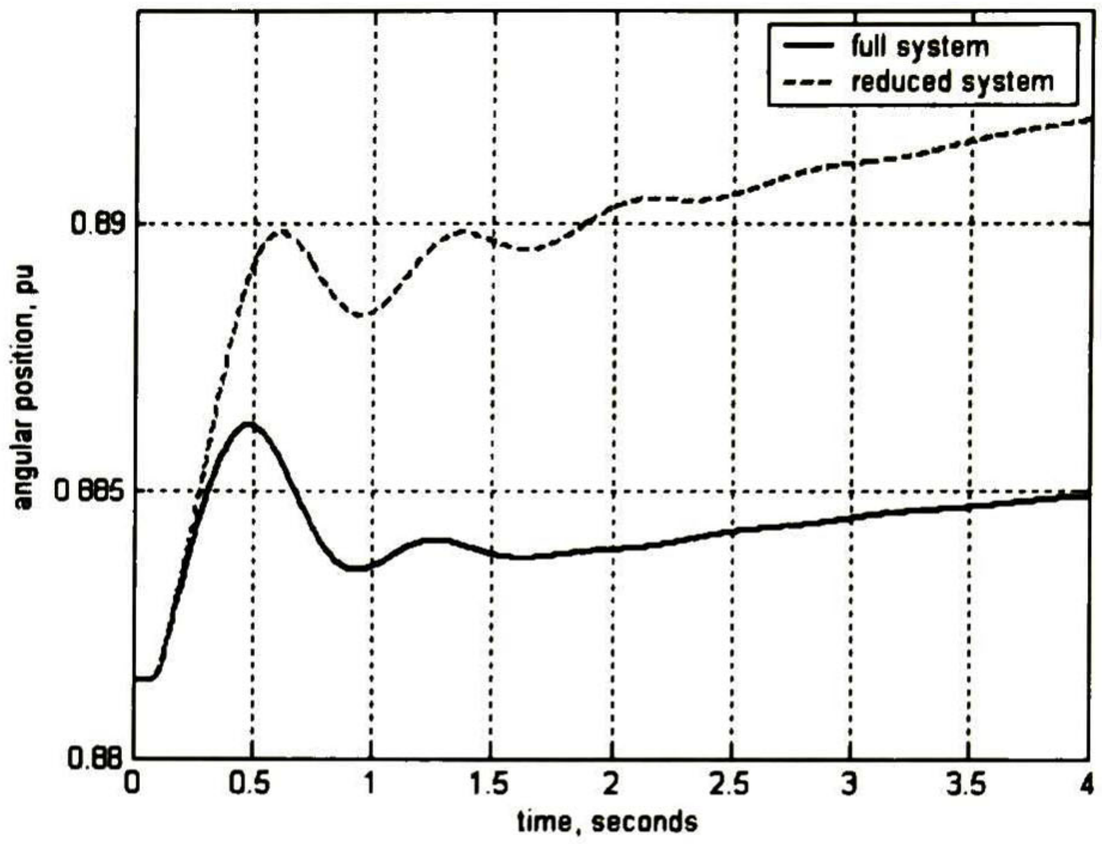



Fig. 4.71 Rotor's machine performance from machine 2 under a load variation at bus 16, case c.

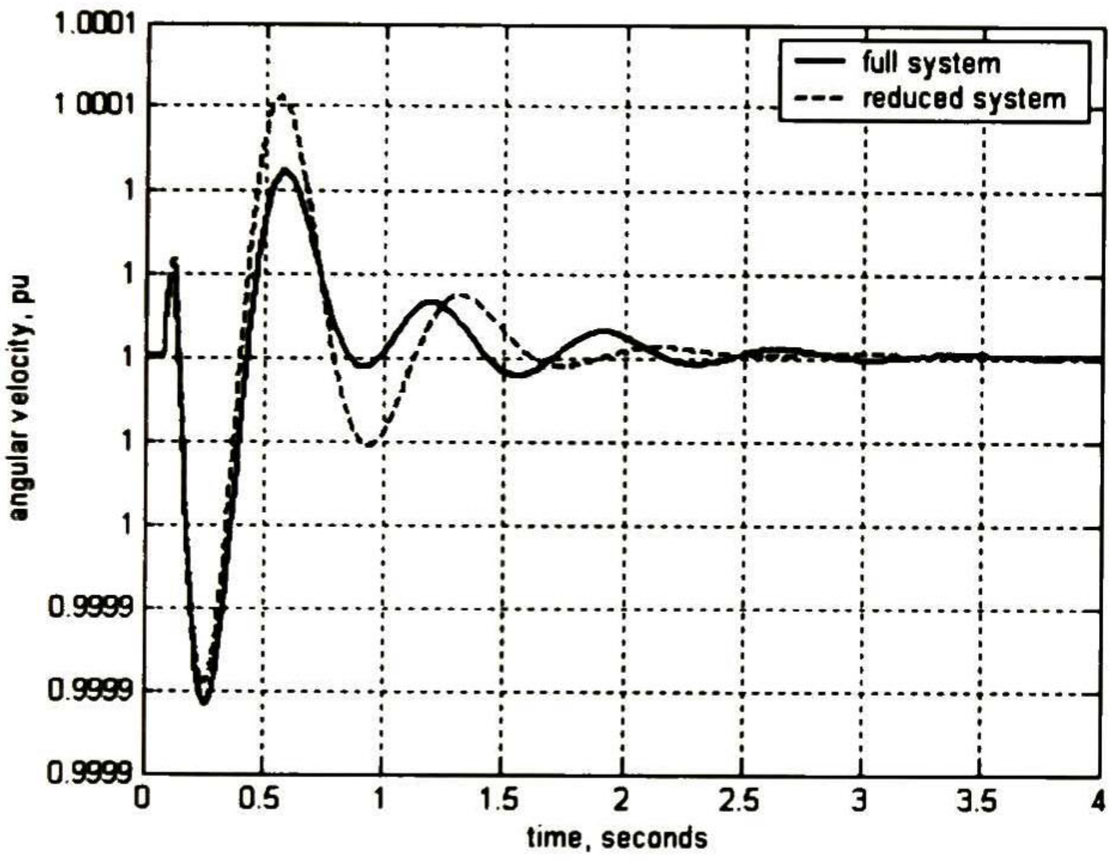



Fig. 4.72 Angular velocity from machine 9 under parameters variation of line 26-28, case c.

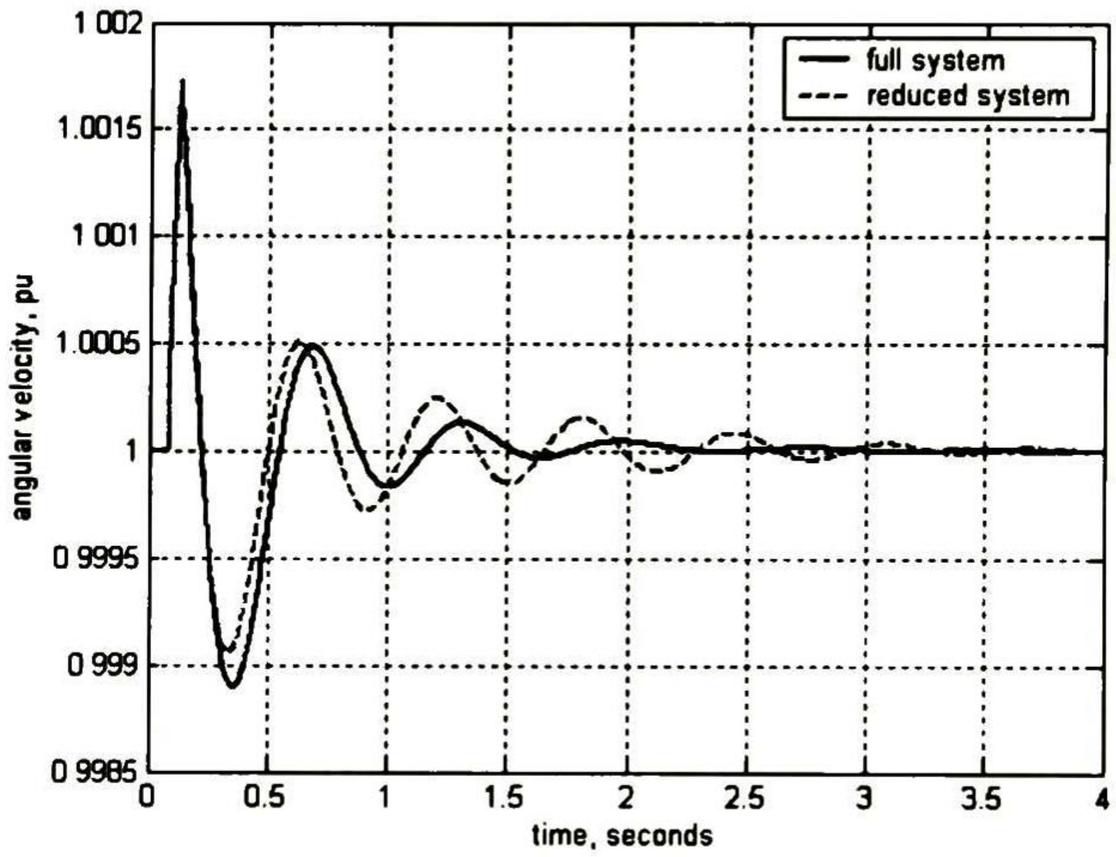



Fig. 4.73 Angular velocity from machine 3 under a 3 phase fault at bus 12, case c.

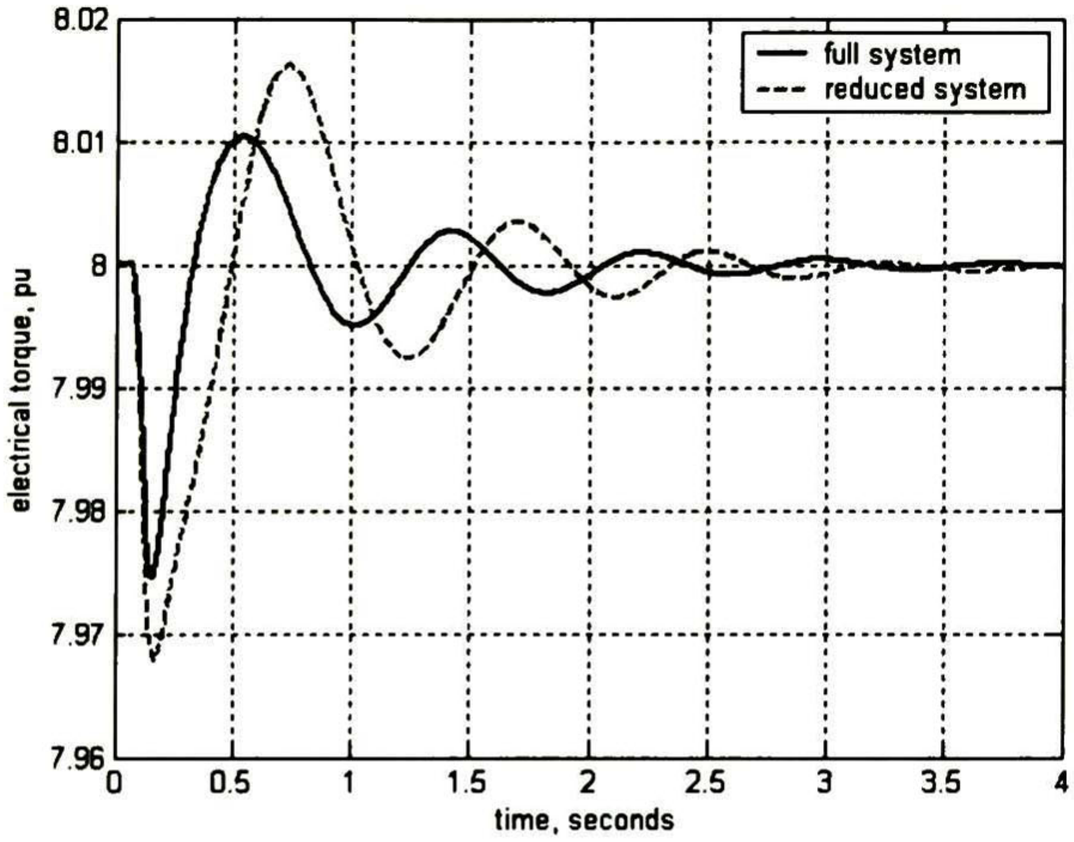



Fig. 4.74 Electrical torque from machine 9 under a load variation at bus 8, case c.

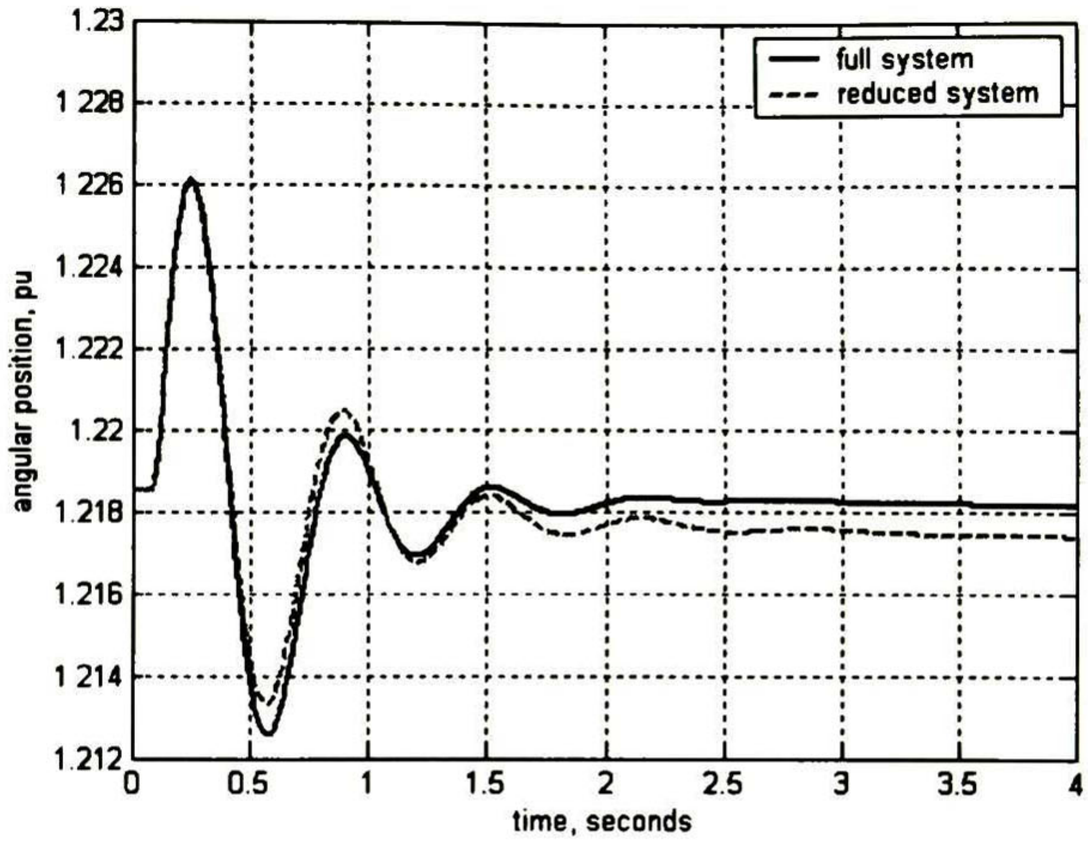



Fig. 4.75 Rotor's machine performance from machine 6 under parameters variation of line 17-27, case c.

From the statistics illustrated in Tables 4.14 and 4.15 once more the robustness, feasibility and confidence of the Dynamic Equivalent is demonstrated. Despite of the RMS errors obtained under these faults which are not as severe as a three phase fault, the information depicted is certainly superior. In both Tables the RMS errors obtained diminished in a considerable mode in comparison with Table 4.14. This is a warranty that the results have been achieved in an excellent way.

**SYNOPSIS.**

In this chapter mainly two methods to construct Dynamic Equivalents are proposed. The first approach is based on a modal preservation, and the feasibly and successfully results show the warranty of this proposed technique.

In the subsequent proposition the influence of novel techniques based on Artificial Intelligence such as Artificial Neural Networks to construct Dynamic Equivalents is developed. This original approach presents great results. The RMS error helps to compute the closeness of time solutions amid full and reduced power system models. The acquired results depict the viability, robustness and confidence of this new technique.

Table 4.15 RMS errors encountered for load variation faults.

| Load on Bus | Machine no. | Case a | | | Case b | | | Case c | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | $\delta$ (degrees) | $\omega$ (rad/s) | Pe (pu) | $\delta$ (degrees) | $\omega$ (rad/s) | Pe (pu) | $\delta$ (degrees) | $\omega$ (rad/s) | Pe (pu) |
| 4 | 1 | 0.3587 | 0.0032 | 0.0133 | 0.4152 | 0.0045 | 0.0059 | 0.4941 | 0.0136 | 0.0211 |
| | 2 | 0.3969 | 0.0035 | 0.0015 | 0.3937 | 0.0036 | 0.0018 | 0.4715 | 0.0085 | 0.0111 |
| | 3 | 0.3974 | 0.0034 | 0.0014 | 0.3926 | 0.0035 | 0.0017 | 0.4688 | 0.0064 | 0.0086 |
| | 4 | 0.3920 | 0.0031 | 0.0007 | 0.3772 | 0.0031 | 0.0007 | 0.4515 | 0.0040 | 0.0018 |
| | 5 | 0.3909 | 0.0031 | 0.0007 | 0.3753 | 0.0031 | 0.0007 | 0.4491 | 0.0041 | 0.0018 |
| | 6 | 0.3945 | 0.0031 | 0.0009 | 0.3794 | 0.0031 | 0.0010 | 0.4529 | 0.0043 | 0.0028 |
| | 7 | 0.3941 | 0.0031 | 0.0007 | 0.3787 | 0.0031 | 0.0007 | 0.4525 | 0.0041 | 0.0018 |
| | 8 | 0.3955 | 0.0031 | 0.0009 | 0.4076 | 0.0036 | 0.0020 | 0.4792 | 0.0086 | 0.0066 |
| | 9 | 0.3937 | 0.0031 | 0.0013 | 0.3751 | 0.0031 | 0.0013 | 0.4731 | 0.0067 | 0.0063 |
| 8 | 1 | 0.3944 | 0.0038 | 0.0136 | 0.4482 | 0.0043 | 0.0051 | 0.5515 | 0.0159 | 0.0253 |
| | 2 | 0.4342 | 0.0046 | 0.0029 | 0.4277 | 0.0048 | 0.0031 | 0.5435 | 0.0184 | 0.0266 |
| | 3 | 0.4339 | 0.0044 | 0.0028 | 0.4256 | 0.0045 | 0.0030 | 0.5354 | 0.0125 | 0.0202 |
| | 4 | 0.4260 | 0.0037 | 0.0012 | 0.4067 | 0.0037 | 0.0011 | 0.5128 | 0.0051 | 0.0034 |
| | 5 | 0.4246 | 0.0038 | 0.0011 | 0.4047 | 0.0037 | 0.0010 | 0.5103 | 0.0054 | 0.0035 |
| | 6 | 0.4288 | 0.0038 | 0.0016 | 0.4092 | 0.0037 | 0.0014 | 0.5147 | 0.0060 | 0.0059 |
| | 7 | 0.4282 | 0.0038 | 0.0011 | 0.4084 | 0.0037 | 0.0010 | 0.5140 | 0.0054 | 0.0036 |
| | 8 | 0.4308 | 0.0036 | 0.0012 | 0.4404 | 0.0038 | 0.0016 | 0.5347 | 0.0099 | 0.0079 |
| | 9 | 0.4279 | 0.0037 | 0.0018 | 0.4043 | 0.0036 | 0.0017 | 0.5285 | 0.0074 | 0.0070 |
| 16 | 1 | 0.2486 | 0.0026 | 0.0134 | 0.2852 | 0.0025 | 0.0021 | 0.3875 | 0.0064 | 0.0081 |
| | 2 | 0.2897 | 0.0024 | 0.0008 | 0.2688 | 0.0023 | 0.0007 | 0.3793 | 0.0043 | 0.0036 |
| | 3 | 0.2904 | 0.0024 | 0.0008 | 0.2681 | 0.0023 | 0.0008 | 0.3776 | 0.0037 | 0.0030 |
| | 4 | 0.2866 | 0.0022 | 0.0006 | 0.2572 | 0.0021 | 0.0004 | 0.3628 | 0.0031 | 0.0011 |
| | 5 | 0.2859 | 0.0022 | 0.0005 | 0.2557 | 0.0021 | 0.0004 | 0.3606 | 0.0031 | 0.0011 |
| | 6 | 0.2885 | 0.0023 | 0.0008 | 0.2588 | 0.0021 | 0.0005 | 0.3639 | 0.0032 | 0.0016 |
| | 7 | 0.2883 | 0.0022 | 0.0006 | 0.2582 | 0.0021 | 0.0004 | 0.3636 | 0.0031 | 0.0011 |
| | 8 | 0.2877 | 0.0023 | 0.0011 | 0.2805 | 0.0023 | 0.0009 | 0.3808 | 0.0049 | 0.0029 |
| | 9 | 0.2874 | 0.0023 | 0.0011 | 0.2557 | 0.0021 | 0.0008 | 0.3764 | 0.0042 | 0.0032 |

**REFERENCES.**

1. De Mello R.W., Podmore R., Stanton K. N., *Coherency-Based Dynamic Equivalents: Applications in Transient Stability Studies*. EPRI, Project RP90-4, Phase II.
2. Lawler J. S. and Schlueter R. A., *Computational Algorithms for Constructing Modal-Coherent Dynamic Equivalents*. IEEE Transaction on PAS No. 5, May 1982: pp 1070-1080.

3. Price W.W, Gulachenski E.M., et al, *Testing of the Modal Dynamic Equivalents Technique*. IEEE Trans. PAS 1978; 97(4): pp. 1366-1372.
4. Belhomme R. and Pavella M., *A composite electromechanical distance approach to transient stability*. IEEE Transaction on Power System, Vol. 6, No. 2, 1990, pp. 622-631.
5. Ramírez Arredondo J. M., *Obtaining dynamic equivalents through the minimization of a line flows function*. Electrical Power and Energy Systems, Vol. 21, 1999, pp. 365-373.
6. Yu Y-N, El-Sharkawi MA., *Estimation of External System Dynamic Equivalent of a Thirteen-Machine System*. IEEE Trans. PAS 1981; 3: pp. 1324-1332.
7. Van Oirsouw P.M., *A Dynamic Equivalent Using Modal Coherent and Frequency Response*. IEEE Transaction on Power Systems, Vol. 5, No. 1, February 1990: pp. 289-295.
8. Undrill J.M. and Turner A.E., *Construction of Power System Electromechanical Equivalents by Modal Analysis*. IEEE PES Winter Power Meeting. Paper 71 TP 136-PWR; pp. 2049-2059.
9. Wang L, Klein M, Yirga S, Kundur P., *Dynamic Reduction of Large Power Systems for Stability Studies*. IEEE Trans. Power Systems 1997; 12(2): pp 889-895.
10. Troullinos G. and Dorsey J., *Coherency and Model Reduction: Space State Point of View*. IEEE Transaction on Power System 1989; 4 (3): 988-995.
11. Di Caprio U., *Conditions for Theoretical Coherency in Multimachine Power System*. 7th IFAC World Congress 1978; 17(5): 687-701.
12. Pires de Souza E., De Macedo N. J.P., et al, *Aplicação de Equivalentes Dinãmicos Baseados em Coerencia em Estudos com Simulador em Tempo Real de Sistemas Electricos*. XIV Seminário Nacional de Producão e Transmissão de Energia Elétrica; Brazil 1997.
13. Chang A. M. M., *Power System Dynamic Equivalents*. IEEE Transaction Vol. PAS-89; Nov-Dec, 1970; pp. 1737-1744.
14. Podmore R., *Identification of Coherent Generators for Dynamic Equivalents*. IEEE Winter Meeting, Paper F77 155-5, New York, 1977.
15. Ganapati P., Sanjay Kumar M., Rasmi Rekha M, *Adaptive Identification of the Parameters of Nonlinear Systems Using Higher Order Statistics*. Conference Proceeding of the ICSPAT-96, Boston, MA, 1996: pp. 200-204.
16. Wehenkel L., *A Statistical Approach to the Identification of Electrical Regions in Power Systems*. Proceeding of Stockholm Power Tech, June 1995: pp. 530-535.
17. Wilfert Hans-Helmut, Voigtländer Knut, Erlich Istvan, *Dynamic Coherency Identification of Generators Using Self-Organising Feature Maps*. Elsevier Control Engineering Practice 2001 (9); pp. 769-775.
18. Djukanovic M, Sobajic DJ, Pao YH., *Artificial Neural Network Based Identification of Dynamic Equivalents*. Electric Power System Res. 1992; 24: 39-48.
19. Wehenkel L, Jacquemart Y., *Use Of Kohonen Feature Maps for the Analysis of Voltage Security Related Electrical Distance*. Proceedings of ICANN'95, International Conference of Artificial Neural Networks, October 1995: pp. 8.3.1-8.3.7.

20. Matsubara T, Nakamura K, Fujita H, Sone M., *Transient Stability Criterion Using Neural Networks.* IEE Japan 2002.
21. J. H. Chow, editor, *Time-Scale Modelling of Dynamic Networks with Applications to Power Systems,* Springer-Verlag, Berlin, 1982.
22. J. J. Moré, *The Levenberg-Marquardt algorithm: implementation and theory.* Watson GA, editor. Numerical analysis, 630. Berlin: Springer, 1977. pp. 105.
23. J. H. Holland, *Outline for a logical theory of adaptive systems.* Journal of the Association for Computing Machinery, 9:297-314, 1962.
24. D. E. Goldberg. *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley, 1989.
25. M. Gen and R. Cheng. *Genetic algorithms & Engineering Design*. John Wiley & Sons, Inc., 1997.
26. Coronado Ixtlahuatl, *Ubicación de Dispositivos FACTS desde una Perspectiva Dinámica,* MSc Thesis, CINVESTAV campus Guadalajara, September 2001.
27. Anderson P. M. and Fouad A. A., *Power System Control and Stability.* IEEE Press, 1993.
28. Kundur Prabha., *Power System Stability and Control,* McGraw-Hill, 1994.

# Conclusions

The use and application of novel techniques which are related to Intelligent Systems have emerged due to the limitations that encompass classic methods in all areas worldwide. These innovative methods have the feature that they are able to adapt by themselves to the problem to be solved even though the application does not posses an exact mathematical model.

In this work an application of one of these methods -Artificial Neural Networks- has been proposed to solve one of the most difficult tasks inside the context of electric power systems, the Dynamic Equivalents. It is presented a brief description that covers the main subjects related to power system stability, and also the principal issues of Neural Networks are explained.

It is concluded that an effective way to construct Dynamic Equivalents is to reduce the complete system merely to a few generators which are labeled as the study system and where the equivalent generators are sited. The propositions have the great advantage of not require the aggregation procedure necessary in classical methods of equivalency.

A model formulation to develop Dynamic Equivalents, where the main goal is to encounter the best estimate parameters for the equivalents generators, is proposed. This technique is posed as an optimization problem without constraints that can be solved by a variety of methods. The minimization algorithms employed, Levenberg-Marquardt and Genetic Algorithms, are ones of the most excellent owing to their robustness and promptly convergence. For all cases where these algorithms are applied, they proved their great features obtaining excellent results. In this methodology genetic algorithms proved to be effective due to the randomness starting point. Through the diversity of operating conditions taken into account, robust dynamic equivalents result.

An improvement to the precedent strategy to calculate Dynamic Equivalents is the use of Artificial Neural Networks. Supported by an Artificial Neural Network some signals from the study system are forecasted. The main advantage of this method is related with the use of typical parameters for the equivalents generators. This technique aids to reduce the considerable computing time associated with transient stability studies of large-scale acquiring outstanding results. A neural dynamic equivalent that does not need assumptions nor linear analysis, as is the case of multiple previous methods is proposed.

The incorporation of the power system stabilizers (PSS) to the tested electric power systems, results to be very important because of they help to verify the feasibility, certainty and robustness of the Dynamic Equivalents. Such inclusion is really important due to it can be considered as an actual

multi-machine power system. The RMS errors corroborate these results. This proposed technique incorporates the limitations that others methodologies can not overtake.

The Artificial Neural Networks attested their great qualities and their huge applications to power systems. The strategies to encounter the right weights such as the preparation of the data set and the validation of the Neural Network under different circumstances demonstrate that the chosen weights are correct. The training procedure is an optimisation algorithm for finding a minimum based on an iterative search algorithm, where the minimum is located by taking a sequence of steps based on the local information connected to the main condition. In this work owing to the achieved results, ANN demonstrate to be a novel tool that can surpass at all the issues related to electric power systems, reducing significantly the overwhelm computing time and cumbersome analysis. Is important to state and to conclude that the input signals chosen for the training process are simply local signals, in other words the power flows are data that can be measure in a real multi-machine power system. This is another advantage to conclude that the Dynamic Equivalent really present robustness and practicability.

Thanks to the excellent acquired results it can be held that the proposed technique is adequate for obtaining Robust Dynamic Equivalents in a good manner reducing the considerable computing time and weighty analysis that characterize the hard task of power systems, taking advantage of the innovative techniques as Artificial Neural Networks.

# Main Contributions

In this dissertation there are two principal contributions, which are:

These two most important contributions are much related to the same issue.
The development of a technique to solve the one of the most difficult problems that encompasses power systems as Dynamic Equivalents is proposed. This technique is proposed to estimate the parameters of fictitious generators that represent dynamic equivalents of an external subsystem. This methodology is proposed an optimisation problem employing different optimisation techniques such as Genetic Algorithms and Levenberg-Marquardt Algorithm. The problem is based on preserve closely those modes highly related with the dynamic of the study subsystem.

Besides, an innovative technique to solve Robust Dynamic Equivalents is proposed. This novel proposition used a new generation of techniques called Artificial Intelligence. In this proposed methodology the application of Artificial Neural Networks (ANN) is employed to solve the hard task of constructing Robust Dynamic Equivalents. The main objective is to create Robust Dynamic Equivalents assisted by an ANN able to reproduce the complex voltage at frontier buses. This proposition presents the advantage of avoiding the aggregation of generators, as in the classical equivalency methods. A neural dynamic equivalent that does not need assumptions nor linear analysis, as is the case of multiple previous methods is proposed.

For both cases, to confirm the accuracy of results, an RMS difference is applied to compare signals and the depicted Figs. showed the feasibility and robustness of these proposed techniques.

# Suggestions for further developments

Next a list of future work that can be done as an extension of this dissertation are described.

1. The application of this proposed technique to a larger multi-machine power system such as the Mexican electric grid.

2. To improve the base data manage from any multi-machine power system in order to develop Dynamic Equivalents.

3. The application of the new generation of neural networks as the Hopfield nets which can be use for the optimisation technique or the self-organising feature maps (SOFM) and to improve the training process applying neural networks that posses unsupervised learning capabilities.

4. The use of other local input signals for the neural network. The current signal for each bus could be an option.

# Publications

These are the publications that have been done during this MSc work.

1. ***"An optimal power system model order reduction technique"*** submitted to *International Journal of Electrical Power & Energy Systems. Elsevier Science Ltd.*

2. ***"Neural Dynamic Equivalents"*** submitted to *International Journal of Electrical Power & Energy Systems. Elsevier Science Ltd.*

3. ***"A technique to reduce power systems electromechanical models"*** accepted. *IEEE PES letters.*

4. ***"Identification of Dynamic Equivalents Preserving the Internal Modes"*** to be presented at *PowerTech 2003, Bologna Italia.*

5. ***"Dynamic Equivalents Based on Artificial Neural Networks"*** submitted to the *North American Power Symposium, NAPS-03, University of Missouri-Rolla, EUA.*

6. ***"Cálculo de Equivalentes Dinámicos Robustos"*** submitted to *Reunión de Verano de Potencia, RVP-03, Acapulco Gro.*


# 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: NEURAL DYNAMIC EQUIVALENTS del(a) C. Rodrigo Joel GARCIA VALLE el día 2 de Julio de 2003 .

DR. JUAN MANUEL RAMÍREZ ARREDONDO
INVESTIGADOR CINVESTAV 3A
CINVESTAV GDL GUADALAJARA

DR. JOSÉ MANUEL CAÑEDO CASTAÑEDA
INVESTIGADOR CINVESTAV 2C
CINVESTAV GDL GUADALAJARA

DR. JUAN MARCOS GARCÍA MARTÍNEZ
PROFESOR-INVESTIGADOR TITULAR B --
UNIVERSIDAD DE GUADALAJARA
--