Artificial Neural Networks for State Estimation of Electric Power Systems

by

Rastko Živanović

Thesis presented for the Degree of

Doctor of Philosophy

in the Department of Electrical and Electronic Engineering

under the supervision of

Prof. Alexander Petroianu

July 1996

Artificial Neural Networks for State Estimation of Electric Power Systems

by

Rastko Živanović

I hereby certify that this is my own original work and has not been submitted before for any other degree.

Signed by candidate ___ JULY 1996

Candidate Date

Recommended for Submission:

_____

Supervisor

_____

Date

Contents

List of Figures

List of Tables

Acknowledgements

I wish to acknowledge the contribution of Professor Alexander Petroianu for his guidance and continuous the support throughout my research and preparation of this thesis.

I would like to express my deepest gratitudes to my wife, Marija, and my mother Milena and my father Miroljub, for their love, patience and support that kept me going throughout my education and professional career.

Terms of References

This thesis originates from the need expressed by ESKOM in 1992 for a faster and more reliable state estimation function in the National Control Centre. The University of Cape Town (UCT) was approached to investigate the possibility of using new emerging technologies, namely Artificial Neural Networks (ANN) and phasor angle measurements, in state estimation. The results of the research on the use of ANN and phasor angle measurements in state estimation are contained in this thesis.

This Ph.D. thesis was commissioned by Professor Alexander Petroianu of the Department of Electrical and Electronic Engineering at UCT in 1993. His specific instructions were:

- To develop theoretical foundations for power system state estimation based on ANN technology. To address the problems associated with the filtering out of small measurement errors, bad data processing, and potential speeding up of the state estimation function, all based on ANN technology.

- To address the problem of using voltage phasor angle measurements in the ANN based state estimation.

- The focus of the thesis should not be detailed practical implementation of the methods developed. The methods in this thesis should form a sound foundation for practical implementation in the future. The methods need to be

tested through simulation by using the available benchmark power systems, for example the IEEE 14-bus system.

Synopsis

This thesis deals with the application of Artificial Neural Network (ANN) technology in power system state estimation. It addresses the following developments:

- the fundamentals of the state estimation based on ANN technology,
- the feasible ANN state estimation methods,
- use of voltage phasor angle measurements in ANN state estimation, and
- bad data processing for ANN state estimation.

The power system state estimation problem is formulated as an optimization problem applied to dynamic ANN model. Two types of dynamic ANN models are used:

- ANN model with steepest descent dynamic, and
- ANN model with Hopfield-style dynamic.

The complexity of an ANN State Estimator (ANN SE) is reduced by using the following techniques:

- a special algebraic transformation of the ANN objective function, and
- the incorporation of zero-injection measurements by the using variable reduction technique.

At the same time, these two techniques improve the filtering performance of the ANN SE.

Two methods for designing the ANN SE for a specific power system are developed:

- an analytical method: it maps the structure and the parameters of a power system into the ANN SE structure and parameters, and
- a synthetic method: it is based on the Real Time Recurrent Learning (RTRL) technique (used in training dynamic ANN), where the ANN SE structure and parameters are determined through learning from available input/output (measurements/state variables) data.

In continuation of the thesis feasible ANN SE methods are developed. It is shown that the well known decoupling technique (used for example in Fast Decoupled State Estimator) can be used in order to create two independent ANN estimators: active ANN SE and reactive ANN SE. This new Decoupled ANN SE can be used in static and tracking modes of operation, as shown in the thesis. The decoupled ANN SE has been tested in a simulation study. The IEEE 14-bus power system was used. The simulation results, supporting the theoretical developments, are presented in the thesis.

Voltage phasor angle measurements are incorporated in the ANN SE. Instead of using absolute bus voltage phasor angle measurements we propose the use of the angle difference between two voltage phasors at adjacent buses. This eliminates the problem with different estimator and absolute angle measurement references. The modified ANN SE was subjected to a sensitivity and simulation study. The results are reported in the thesis. It was found that phasor angle measurements improve the accuracy of state estimation if they complement, not replace, power flow and injection measurements.

A bad data processing technique for the ANN SE was developed. The new technique is based on the RTRL with the Least Absolute Value (LAV) objective function. It can efficiently eliminate the influence of bad measurements, bad parameters, or wrong topology on the ANN SE results. The results of a simulation study conducted with this new technique are reported in the thesis.

Nomenclature

The following is a list of the main symbols used in this thesis.

$\mathbf{Z}$	Measurement vector
$\mathbf{X}$	State vector
$\mathbf{h(X)}$	Non-linear measurement vector function
$\mathbf{v}$	Measurement errors
M	Dimension of measurement vector $\mathbf{Z}$
N	Number of nodes in network
n	Dimension of state vector $\mathbf{X}$
$\mathbf{R}$	Covariance matrix
$\mathsf{E}[\cdot]$	Expectation value of $[\cdot]$
$\mathbf{H}$	Jacobian matrix
SE	State Estimator
LS	Least Squares
LAV	Least Absolute Value
ANN	Artificial Neural Network
$\mathbf{W}$	Weight matrix of ANN
$\mathbf{U}$	Input vector of ANN
τ	Time constant
μ	Gain constant
WLS	Weighted Least Squares

$\mathbf{S}$	Sensitivity vector
$\bar{S}$	Sensitivity measure
var $[\cdot]$	Variance of $[\cdot]$
RTRL	Real Time Recurrent Learning
CTRNN	Continuous Time Recurrent Neural Network
DTRNN	Discrete Time Recurrent Neural Network
$\mathbf{I}$	Identity matrix
δ	Kronecker delta
σ	Standard deviation
V_i	Voltage magnitude
θ_i	Voltage phasor angle
P_{ij}	Active power flow
P_i	Active power injection
Q_{ij}	Reactive power flow
Q_i	Reactive power injection
k	Iteration counter
q	Sample counter
L_u	Upper limit of tanh activation function
L_l	Lower limit of tanh activation function
c	Slope constant of tanh activation function
PI	Performance Index
FDSE	Fast Decoupled State Estimator
PMU	Phasor Measurement Unit
GPS	Global Positioning System
VT	Voltage Transformer
CT	Current Transformer

DFT	Discrete Fourier Transform
$\mathbf{r}$	Measurement residual vector
$\mathbf{P}$, $\mathcal{W}$, and $\mathbf{\Gamma}$	Sensitivity matrices

Chapter 1

Introduction

Modern electrical power systems consists of large synchronous generators and concentrated load centres interconnected over large geographical distances. In such an interconnected system power generation has to track the randomly varying load. As a result, real-time operation of a power system becomes an increasingly complex task. Real-time control of a power system is performed in control centres through computerised control functions, so called Energy Management System (EMS). In general the role of an EMS is to minimize the production cost while maintaining the system in a normal and secure state [1],[2]. These two goals are achieved by the application of security and optimization functions as a part of EMS [3],[4],[5]. The input for these functions is the State Estimator (SE) output. SE is responsible for providing a reliable data base from the raw data measured throughout the power system. The application of state estimation in power system control was first suggested in 1970 by Prof. Fred Schweppe and his research team from MIT [6].

1.1 An Overview of State Estimation Function

A State Estimator algorithm calculates the state of a power system using the information on the status of breakers/isolators and on analog measurements, such as bus voltages, active and reactive power flows, and generator/load active and reactive power injections [6], [7]. It also uses the status of transformer/phase shifter

tap positions, and in some recent cases voltage and current phasor measurements [8]. Measurements and status data are collected during cyclic scans of remote terminals performed by the Supervisory Control and Data Acquisition (SCADA) system [7]. The active and reactive powers of zero injection buses (those without generators and loads) are also inputs to the SE. They are treated either as measurements of high accuracy, or as explicit equality constraints [9].

The output of a SE algorithm is the state vector $\mathbf{X}$ which contains the bus voltage phasors, magnitude V and angle θ, at all the buses except one bus which is used as a reference for the angles. The angle of the reference bus is arbitrarily excluded from the state vector. Another possible option is to use an external (fictitious) reference, i.e. all angles are included in the state vector [10].

The performance of a SE algorithm depends on the number, location, and quality of the measurements, and hence on the quality of the instrument transformers, transducers, analog/digital converters, and filters employed [11]. The quality of the measurements can be expressed in terms of variance or confidence factors that are used to weight the measurements with respect to each other.

Role: The role of a state estimation function can be summarized as follows [12]:

- to provide an estimate for all metered and unmetered electrical quantities of the power network,
- to filter out small errors which corrupt the metered values in order to compute estimates that satisfy Kirchhoff's laws,
- to clean the data base of all bad data.

Assumptions: It is very important to understand that the state estimation is based on the following assumptions:

- the network is balanced and the three-phase voltages and currents are sinusoidal; a single phase π-equivalent circuit model of lines and transformers is used,
- the parameters of the equivalent network circuits, such as resistance, reactance, and susceptance are exactly known,
- the real-time topology of the network can be inferred from the available status data of breakers and isolators,
- the measurements are taken simultaneously so that they constitute a "true" snapshot of the state of the power system,
- the analog measurements are corrupted with errors represented by independent Gaussian random variables with zero means, and a known covariance matrix.

Observability requirements: To obtain a SE solution, the number of unknown state variables must be matched by an equal number of independent (nonredundant) measurements. Any area of the power system that satisfies this condition is said to be observable. Additional measurements provide redundancy, in which case the independent measurement set is not unique. Without redundancy, the independent measurement set is said to be critical, because the corresponding equations become undetermined by the loss of just one of these measurements. Detailed explanations of power system observability may be found in [13],[14].

Unobservable areas can be made observable by introducing pseudo measurements. The weights assigned to pseudo measurements are adjusted to reflect their quality relative to telemetered measurements. Pseudo measurement sources include previous estimates, manual entries, and forecasts or scheduled values [11]. The forecasts may be based on factors that relate unknown quantities to known quantities. For example, in the case of a tapped transmission line, pseudo measurements are

values of the loads that are unmetered but considered as fixed percentages of the flow into or out of the line. Another example is the application of load distribution factors to forecast bus loads from known area loads [15]. These bus loads can be used as pseudo measurements. The generation schedules for each control area in the network are generally obtained by executing an economic dispatch. These scheduled values can also be used as pseudo measurements.

Redundancy requirements: Measurement redundancy improves the quality of state estimation. Ideally, redundancy should be distributed uniformly throughout the observable area. It reduces the number of pseudo measurements and it is the key to bad data detection and identification. It is important to differentiate between

- global redundancy, $\eta = M/n - 1$ (where M is total number of measurements and n is total number of states), and
- local redundancy [16].

Global redundancy $\eta > 0$ is a necessary but not a sufficient condition for the achievement of a reliable state estimation. For a reliable state estimation sufficient local redundancy for each measurement must also be considered. Sufficient local redundancy for each measurement is a must for the treatment of bad data [16].

Measurement Errors: The errors in power system data can be classified into two categories:

- small errors which are always present, due to meter inaccuracies, non-simultaneous sampling, parameter deviations, model imperfections and approximations, etc.;
- large errors in the measurements, parameters, and network topology; such errors are due to instrument failures, data transmission errors, incorrectly

wired meters, bad calibration, inconsistent data base representations, falsely transmitted status of switches (breakers and isolators), etc.

Bad measurements are removed or suppressed using a variety of techniques [17]. In bad data processing the validity of the collected data is first assessed through simple plausibility checks. Here, all the measurements beyond the physical limitations of the equipment are labelled as false, and as such rejected from the data base. If the metered value conflicts with the switching status, or if it does not comply with Kirchhoff's laws, it is labeled as a suspected bad measurement. Bad data which passed the plausibility checks undetected, will be detected and identified by the SE function.

Static SE versus Tracking SE: In the present EMS the SE software is running every few minutes and/or is started by any change in the network topology. This type of SE is called static because it processes only one sample of measurements at a time (snapshot). Static SE computation does not permit continuous tracking of all changes in the power system, which may be relevant to real-time monitoring and control. As an addition to static SE, tracking state estimation algorithms were suggested [18]. They produce an estimate for each measurement scan obtained from SCADA, and they filter out small measurement errors more efficiently. Some tracking algorithms include state forecasting which can help in creating pseudo measurements and the treating of bad data [19].

1.2 Related Research

Knowledge on power system state estimation and related problems has reached a high level. Even so many problems are only partly solved and still need further investigation. The research efforts are focused on: decrease of estimation response

time, bad measurements and parameters processing, and development of tracking estimation techniques.

Response Time: The response time of static state estimation is reduced to 5 or even 2 minutes [12]. The periodicity of the data acquisition scan cycles is however in the order of seconds, so that the faster state estimation is needed. The following faster computation techniques can shorten the execution cycle of SE:

- parallel processing [20], and
- Artificial Neural Networks [21],[22].

Then it would be preferable to apply the limit-checking function, which detects incipient or actual emergency conditions, not to the raw measurements but to the SE results [12]. The security monitoring performed on filtered SE result will be more accurate than if the raw measurements were used.

Processing of Bad Measurements: The processing of bad measurements is still a very active research topic. The most advanced methods proposed for Weighted Least Squares (WLS) SE are:

- combinatorial optimization [23], and
- Hypothesis Testing Identification (HTI) [24].

The main weakness of these methods is that they make use of detection tests based on the least squares residuals. These residuals are prone to the masking effect of multiple bad measurements.

A completely different approach in bad measurement processing is the application of non-quadratic norms. Such techniques are proposed in [25],[26]. The Least Absolute Value (LAV) estimator, first suggested in [27], deserves special attention. The most attractive advantage of the LAV SE is in the automatic rejection of bad

measurements, i.e. it does not need any statistical detection test. A number of ideas for improving the basic LAV technique are proposed in [28],[29],[30],[31],[32]. Unfortunately, the LAV estimator becomes unreliable when the multiple bad measurements correspond to the so called bad leverage positions in an estimator factor space [33]. A reweighting strategy for the LAV estimator is proposed to solve the problem caused by bad leverage points [34],[35]. Another approach is to use a Least Median of Squares (LMS) estimator [36]. The LMS estimator is able to cope with a large number of bad leverage points but the computation time is prohibitively long. All of these robust techniques do not perform as well as the WLS technique when the filtering of small measurement errors, modelled as Gaussian noise, is considered. This is why the robustification of the WLS algorithm is proposed [37]. This method identifies the leverage points and computes the projection statistic of each point, comparing it to a statistical cutoff value. Based on these projection statistics, a robustly weighted M-estimator can be defined [37]. It is computed by the algorithm which is a simple modification of the conventional WLS iterative method.

Processing of Bad Parameters: One proposed scheme is to augment the state vector with all unknown (or uncertain) parameters [38]. This approach increases the dimension of the unknowns in the state estimator and results in an increase in computational effort. The observability with respect to the augmented state vector could be a constraint in those areas where insufficient data is available. The method is based on a single measurement sample and it is of limited accuracy. When seeking an accurate parameter estimate, better results can be obtained by using several measurement vectors taken at different times. One such approach is based on Kalman filtering [39]. It consists of processing measurement vectors sequentially. The method uses as an input the parameter vector initially estimated. It can be viewed as an adaptive filter, since the parameter vector estimate improves as the number of measured samples increases. This improvement results in a decrease of

the variance of the parameter vector estimate. The third and the most promising approach is based on the residuals obtained through SE and used for processing bad measurements[40]. The method is based on the sensitivity relationship between the measurement residuals and the parameter error. This relationship is treated as an estimation problem, yielding a parameter error estimate. Dimension and observability problems are not critical in this approach. This method can be applied to single or several measurement samples.

Tracking Estimator: Another line of research is directed towards the application of dynamic state estimation theory in power systems [41]. One application is parameter estimation by using Kalman's filtering theory which has already been mentioned. We can also estimate the state vector by using the Kalman filter or some other form of adaptive filter. The most important benefit which could be obtained is state prediction. Such an estimator provides the necessary information to perform preventive network analysis and control [42]. It also provides pseudo measurements which can be used for observability improvement and bad measurement processing [19],[43].

Since the application of the ANN techniques in state estimation is the objective of this thesis, the overview of the relevant theoretical and practical results obtained in ANN related research is presented in the next section.

1.3 Artificial Neural Networks

1.3.1 State-of-the-Art

Let us first briefly summarize the research results on the capabilities and limitations of the ANN models which can be used in power systems. An ANN model consists of a number of neurons, that constitute the elementary processing units.

Neurons are connected together in a set pattern [44]. Each neuron can receive several inputs. These inputs are weighted by means of parameters (weights). The inputs are combined to give a net input to the neuron. The net input is further processed by an activation function (linear or non-linear) to obtain the neuron output. There are two basic categories of ANN models [45]: static (feed-forward) and dynamic (feed-back).

Static ANN models, of which the multilayer perceptron is the most widely used in power system applications, are characterized by node equations that are memoryless. That is, their output is a function only of the actual input, not of past or future inputs or outputs. These networks are useful in a variety of applications such as function approximation and pattern recognition. The learning of such networks is usually accomplished by using a Back Propagation algorithm [46],[47].

Dynamic ANN models, on the other hand are the systems with memory. Their node equations are described by differential or difference equations. Dynamic and resonance characteristics of these networks make them the appropriate tool in modelling and controlling processes in non-linear dynamic systems, such as power system. The potential areas of application of such ANN models are in adaptive control and optimization. The ANN models with steepest descent dynamic and with Hopfield-style dynamic are probably the best known dynamic network models [48], [49]. All of these networks consist of a single layer of neurons that are fully interconnected. It is possible to demonstrate that the discrete-time approximation of the Hopfield-style ANN [50] can be reduced to an ordinary multilayer perceptron [45]. Back Propagation Through Time is one possible algorithm for learning dynamic ANN [47]. Here, the ANN is unfolded over time, and a Back Propagation algorithm is applied. Real Time Recurrent Learning (RTRL) is the another option [49],[50], where the learning of ANN is performed recursively.

In the last few years the hardware implementation of ANN models reached a mature status [51],[52],[53],[54], and even became commercially available [55]. This special hardware enables the real-time solution of large control and optimization problems.

1.3.2 Power System Applications

The classical approach in solving a power system problem involves a detailed study leading to a mathematical model, and then to the development of suitable methods for solving the model. On the other hand, ANNs are trained to solve the problem by learning from examples. The field is particularly exciting today because ANNs can be implemented by using special purpose hardware [55].

Over the past few years, the application of different ANN computation techniques has been proposed for a wide spectrum of tasks in power system operation [56]. Estimation of critical clearing time using a Back Propagation algorithm in a feed-forward ANN was one of the first breakthroughs in the field [57]. Other applications include [58]: security assessment , load forecasting , fault locating, state estimation, alarm processing, optimal capacitor switching, power system stabilizer tuning, etc. The feed-forward ANN application in state estimation is proposed in the following two papers.

Kanekar at al. [64] use several independent random noise sources for measurement error simulation in order to provide the sets of inputs for the ANN estimator. The Kalman filter is used to find estimates for each of the input measurement sets. The average of these estimates is used in training as the desired output for the ANN estimator. It is shown in the paper that the trained ANN estimator gives the accurate state estimates when the measurements are subjected to unknown noise.

The pattern analysis technique has been developed for solving power system state estimation problem [65]. The technique is based on feed-forward multilayer ANN.

The advantages of the technique are: it provides the accurate pseudo measurements, and it can detect and identify bad data in the situations when the methods based on measurement residuals will always fail. The shortcoming of this technique is the need for extremely large training sets in order to produce reliable estimates for all possible situations; the consequence is the prohibitively long training time. The authors propose network decomposition and fast training algorithms to overcome this problem.

Beside the feed-forward ANN, neural networks with feed-back form the next class of computation techniques employed in solving power system problems; for example unit commitment and dispatch problems [59],[60], load flow and optimal power flow [61],[62],[63], as well as the state estimation [21],[22].

Nakagawa at al. [21] propose a Hopfield-style ANN (ANN with state feed-back) application to the state estimation computation. They have compared computing times of the traditional WLS estimator and the Hopfield-style ANN estimator. The results show that the traditional estimation technique is faster than Nakagawa's ANN.

Živanović at al. [22] elaborate the Nakagawa's idea introducing the approximation of the Hopfield-style ANN estimator. The approximation makes the synapse weights of the ANN estimator constant, for one structure of a power system. They show that making the weights constant will shorten the execution time of the ANN estimator. Also they have compared the filtering performance of the traditional WLS estimator and the ANN estimator. The results indicate the advantage in using the ANN estimator. This thesis elaborates further on the ideas proposed in this paper.

1.4 Outline of the Thesis

This thesis is divided into six chapters. Each of these chapters contains the main results required for understanding the thesis. The thesis is supplemented with four appendices. These appendices contain information that are required for detailed analysis of certain contributions, and data of the power systems used in the simulation studies. The following provides a brief outline of the thesis.

At the beginning of Chapter 2 we present an overview of the power system measurement model used in state estimation. After that the Least Squares method is discussed. It represents the foundation for existing estimation algorithms as well as for new developments.

Further on, in the same Chapter, the theoretical fundamentals of an Artificial Neural Network State Estimator (ANN SE) are developed. Two versions of the ANN SE are proposed: ANN SE with steepest descent dynamic and ANN SE with Hopfield-style dynamic. Then, we address the problem of designing ANN SE for a specific power system. Two approaches in design are proposed: analytic and synthetic. The analytic design directly maps the parameters and the structure of a measurement model into ANN connection weights. In the synthetic approach Real Time Recurrent Learning (RTRL) technique is applied for learning the connection weights of the ANN SE.

Virtual measurements are incorporated as equality constraints in the ANN SE. In that manner, the constrained ANN SE is formulated. Further on, the algebraic transformation of an ANN SE objective function is introduced, creating the new ANN SE structure. Both modifications reduce the ANN SE implementation cost, and improve the performance.

At the end of Chapter 2 we address the implementation of the ANN SE.

In Chapter 3 we describe a practically feasible ANN SE method: Decoupled ANN SE. Then we use Decoupled ANN SE for static and tracking state estimation. The method is tested in a simulation study. The results obtained from the simulation study are presented with discussion, at the end of the Chapter.

In Chapter 4 we are investigating the effects of additional voltage phasor angle measurements on the ANN SE performance. In using absolute voltage phasor angle measurements directly in the state estimation algorithm we need to solve the problem of different angle references. Namely, the voltage phasor angle for one bus is chosen as a reference in state estimation while the reference for absolute angle measurements depends on how the sampling is initiated on specific instrument. Two ways of solving this problem are proposed in Chapter 4.

Subsequently, we present an approach where the angle difference between two voltage phasors at adjacent buses (angle shift) is used instead of the absolute bus voltage phasor angle measurement. In this approach the reference problem does not exist. Simulation and sensitivity studies have been devised in order to demonstrate the advantages of the ANN SE with additional angle measurements. In Chapter 4 we present the results and a discussion of the studies.

In Chapter 5 we describe a new method for bad data processing developed specially for the ANN SE. The method is based on the RTRL technique. This technique has been used for ANN SE synthesis in Chapter 2. The RTRL technique is modified so that the Least Absolute Value (LAV) objective function is used instead of Least Squares (LS) objective function. The RTRL technique adapts the weights of ANN SE, and in this manner automatically rejects the influence of bad data on state estimate. The technique is used in treating gross errors in: measurements, parameters and topology. The results of the case studies conducted with this technique are presented in the chapter. At the end of Chapter 5 we analyze the properties of the proposed technique, and we identify the potential drawbacks.

Chapter 6 concludes the thesis. It discusses the contributions proposed in the thesis. This chapter ends with suggestions for future work.

Appendix A presents a detail derivation of the power system model used in state estimation. This model forms the basis for developments in Chapter 2. This Appendix also presents the simplified power system model. This model is used in Chapter 3 for development of Decoupled ANN SE.

Appendix B provides the derivation of the formulas for calculating the sensitivity measures of the ANN SE in static and tracking modes of operation. We use these sensitivity measures to assess the filtering performance of the ANN SE. These sensitivities are very useful in evaluating the different sets of weights obtained in the synthetic design of ANN SE. These weights can provide the same input-output mapping but they will exhibit different sensitivities to measurement errors.

Appendix C presents a complete derivation of the Real Time Recurrent Learning (RTRL) technique for ANN SE. This technique is used in Chapter 2 for the synthetic design of the ANN SE, and in Chapter 5 for bad data processing.

Appendix D provides data for 3-bus and 14-bus power systems used in the simulation studies throughout the thesis.

Chapter 2

ANN State Estimator: Fundamentals

In this chapter the power system state estimation is formulated as an optimization problem applied to dynamic ANN model [45]. The dynamic ANN is in general a large-scale system defined by a set of differential equations together with an appropriate objective function. The optimization process consists of starting from an arbitrary point and allowing the dynamic ANN to converge to a stable point corresponding to a minimum of the objective function. The equilibrium state of the dynamic ANN represents the solution of the optimization problem, and at the same time the state estimate for a given set of measurements.

Two different approaches in the design of an ANN State Estimator (ANN SE) are used in this chapter:

- Analytic design: The structure and parameters of an ANN SE are determined directly from a model relating measurements to the power system state variables (i.e. nonlinear measurement model). The measurement model has to be linearized. The assumption that errors in the measurement model are independent (not correlated), and normally distributed random numbers should be complied with.

- Synthetic design: The parameters for a specified structure of an ANN SE are obtained by training it either on simulated data or in real-time operation. In

the synthetic design we do not need any of the assumptions which are required for the analytic design [66].

The fundamentals of two ANN SE methods are developed in this chapter:

- The first method is based on the ANN with steepest descent dynamic [53],[67], and
- the second one is based on the ANN with Hopfield-style dynamic [45].

Both design approaches can be used for steepest descent ANN SE. In the design of Hopfield-style ANN SE only the synthetic approach can be used. Special technique, so called Real Time Recurrent Learning (RTRL) [49],[50], is used in this chapter for the development of the method for synthetic design.

The complexity of an ANN SE can be reduced by an appropriate algebraic transformation of the least squares objective function [68], and by the incorporation of virtual measurements. The proposed algebraic transformation yields to the augmented form of ANN SE. The virtual measurements are incorporated using the variable reduction technique [67],[69]. The fundamentals of both techniques are presented in this chapter.

2.1 The Measurement Model

The state of an electrical power network is determined by the n dimensional vector $\mathbf{X}$ ($n = 2N - 1$, where N is the total number of nodes). The elements of the state vector $\mathbf{X}$ are the magnitudes and angles of the voltage phasors associated with each busbar of the power network. An estimate of the state vector $\mathbf{X}$ can be computed from the system data consisting of network structural information, system parameter values, and a sufficient set of measurements.

The M dimensional measurement vector ($M > n$) is denoted by $\mathbf{Z}$, and modelled as [15]

$$\mathbf{Z} = \mathbf{h}(\mathbf{X}) + \mathbf{v}, \tag{2.1}$$

where $\mathbf{h}(\mathbf{X})$ is a nonlinear vector function, mapping the state vector $\mathbf{X}$ into the measurement vector $\mathbf{Z}$. The power network equations represented by $\mathbf{h}(\mathbf{X})$ are given in Appendix A.

The vector consisting of the measurement errors is denoted by $\mathbf{v}$ in the Eq. (2.1). For the actual meter readings, these errors are the result of high frequency noise, drift, calibration errors, analog to digital conversion errors etc. [70]. In this study, errors are assumed to be time invariant, independent random variables with normal distribution with zero mean. The covariance matrix of the measurement errors is constant and it is denoted by $\mathbf{R}$, i.e.

$$\mathbf{R} = \mathsf{E}\left[\mathbf{v}\mathbf{v}^T\right] = \begin{pmatrix} \sigma_1^2 & & 0 \\ & \sigma_2^2 & \\ & \ddots & \\ 0 & & \sigma_M^2 \end{pmatrix} \tag{2.2}$$

where σ_i^2 is the variance of the measurement error v_i.

In reality the covariance matrix $\mathbf{R}$ may actually be time varying because the errors can be dependent on the measured values [18]. It is obvious that for the static state estimation concept based on only one sample of measurements, the time structure of the measurement errors is not important, therefore it can be assumed that the matrix $\mathbf{R}$ is constant. But, the time structure of errors is very important when considering a tracking (dynamic) state estimation concept [71]. The reason being that a tracking SE is using more than one sample of measurements to obtain an estimate. Two extreme types of time structure are the white sequence type (or Gaussian noise) where $\mathbf{v}(t)$ is independent of $\mathbf{v}(\tau)$ for $t \neq \tau$, and the biased type

where $\mathbf{v}(t) = \mathbf{v}(\tau)$. In real applications the time structure will lie between these two extreme cases [71]. For the white sequence of errors it is possible to design a tracking estimator which performs "smoothing" over time (i.e. the measurement error varies randomly while the state varies smoothly). Such an estimator has a time constant which allows more data to be averaged for one state estimate. For biased errors, such smoothing has no effect unless an attempt is made to estimate the bias. Departure from the white sequence type of errors can be eliminated through metering system calibration [70].

2.2 The Least Squares Method

The Least Squares (LS) Method gives the optimal state estimate (in the sense of minimum variance of the estimation error) if the measurement errors are represented by the white sequence type [71].

Let us first consider the static state estimation (time-invariant) concept. Assume that one snapshot of measurements is available at specific time t_q and let $\mathbf{Z} = \mathbf{Z}(t_q)$, $\mathbf{X} = \mathbf{X}(t_q)$, and $\mathbf{v} = \mathbf{v}(t_q)$. The optimal estimate $\hat{\mathbf{X}}$ is given by the value of $\mathbf{X}$ for which the scalar sum of the weighted squares of measurement residuals (the objective function)

$$J(\mathbf{X}) = \frac{1}{2}[\mathbf{Z} - \mathbf{h}(\mathbf{X})]^T \mathbf{R}^{-1} [\mathbf{Z} - \mathbf{h}(\mathbf{X})] = \frac{1}{2} \sum_{i=1}^{M} \frac{(z_i - h_i(\mathbf{X}))^2}{\sigma_i^2}, \tag{2.3}$$

reaches its minimum value [72],[73]. Since the estimate $\hat{\mathbf{X}}$ minimizes the square error weighted by the measurement accuracy, it is commonly called the Weighted Least Squares (WLS) estimate [71]. The solution of WLS problem is obtained by using iterative techniques applied to the linearized error function (2.3). The linearized and simplified objective function (2.3) is, [73]:

$$J(\Delta\mathbf{X}) = \frac{1}{2}[\Delta\mathbf{Z} - \mathbf{H}\Delta\mathbf{X}]^T [\Delta\mathbf{Z} - \mathbf{H}\Delta\mathbf{X}], \tag{2.4}$$

where

$$\Delta \mathbf{X} = \mathbf{X}_{k+1} - \mathbf{X}_k, \tag{2.5}$$

$$\Delta \mathbf{Z} = \mathbf{Z} - \mathbf{h}(\mathbf{X}_k), \tag{2.6}$$

and

$$\mathbf{H}(\mathbf{X}_k) = \left. \frac{\partial \mathbf{h}(\mathbf{X})}{\partial \mathbf{X}} \right|_{\mathbf{X}=\mathbf{X}_k}, \tag{2.7}$$

is the Jacobian matrix calculated for $\mathbf{X}_k$. $\mathbf{X}_k$ is the solution at the iteration step k. The formulas for calculating the entries of the Jacobian matrix (2.7) are given in Appendix A. In order to simplify the notation, we redefined $\Delta \mathbf{Z}$ as $\mathbf{R}^{-\frac{1}{2}} \Delta \mathbf{Z}$ in the Eq. (2.4). Similarly, the Jacobian matrix $\mathbf{H}(\mathbf{X}_k)$ is redefined as $\mathbf{R}^{-\frac{1}{2}} \mathbf{H}(\mathbf{X}_k)$. Hence, it is not necessary to explicitly consider matrix $\mathbf{R}$ in equation (2.4).

In order to minimize the objective function $J(\mathbf{X})$ given in (2.3) a Gauss-Newton iterative algorithm [69],[74] is commonly used

$$\mathbf{G}(\mathbf{X}_k)(\mathbf{X}_{k+1} - \mathbf{X}_k) = \mathbf{H}^T(\mathbf{X}_k)(\mathbf{Z} - \mathbf{h}(\mathbf{X}_k)), \tag{2.8}$$

where

$$\mathbf{G}(\mathbf{X}_k) = \mathbf{H}^T(\mathbf{X}_k)\mathbf{H}(\mathbf{X}_k) \tag{2.9}$$

is the gain (information) matrix. At each iteration step the linearized quadratic error function $J(\Delta \mathbf{X})$ (given by Eq. (2.4)) is minimized. For each iteration the inversion or L\U factors of the gain matrix $\mathbf{G}(\mathbf{X}_k)$ must be found [69],[74]. The ANN can be used in solving the linear system of equations (2.8) [22]. This approach is very similar to the Gauss relaxation method [74].

When the algorithm converges, the estimate $\hat{\mathbf{X}}$ minimizes $J(\mathbf{X})$, and $J(\hat{\mathbf{X}})$ is the minimum of weighted sum of the squared measurement residuals for one snapshot of measurements. The level of confidence in the static state estimator result is given by the covariance matrix of the estimation error $\mathbf{X}^* - \hat{\mathbf{X}}$ (derived in Appendix B)

$$\mathsf{E}\left[(\mathbf{X}^* - \hat{\mathbf{X}})(\mathbf{X}^* - \hat{\mathbf{X}})^T\right] = \left(\mathbf{H}(\hat{\mathbf{X}})^T \mathbf{H}(\hat{\mathbf{X}})\right)^{-1}, \tag{2.10}$$

where $\mathbf{X}^*$ is the true state vector.

Now we look beyond the static estimation concept into tracking state estimation where the application of ANN technology could exhibit the major benefits over classical techniques. The objective function to be minimized in tracking state estimation is the mean square error denoted by $\mathsf{E}[J(\mathbf{X})]$, where $J(\mathbf{X})$ is given in (2.3). We assume that the measurements are statistically stationary (measurement samples are available at a much higher rate than the rate of change of the system state), and we linearize $\mathsf{E}[J(\mathbf{X})]$. The linearized objective function is a quadratic performance surface around the point of linearization [75]. Actually the performance surface is smoothly moving along with the daily load cycle. The location of points on the surface can be estimated by averaging the linearized square error $J(\Delta\mathbf{X})$ over a period of time. The methods for searching the minimum of $\mathsf{E}[J(\mathbf{X})]$ entail the use of gradient estimates (instead of true gradient) to indicate the direction in which the minimum of the surface lies [75]. So instead of calculating the estimate for one snapshot of measurements, like in static SE, we use in the tracking SE an average over a number of snapshots. In this manner, we can more effectively filter out normally distributed errors whose mean is zero. Practical implementations can be batch or recursive data processing [71].

2.3 The ANN State Estimator with Steepest Descent Dynamic

In the ANN technology the minimization of the linearized objective function $J(\Delta\mathbf{X})$ (given by Eq. (2.4)) can be performed using the steepest descent dynamic process associated with the system of differential equations [76],[53]

$$\tau_x \frac{d(\Delta\mathbf{X})}{dt} = -\nabla_x J(\Delta\mathbf{X}), \tag{2.11}$$

where τ_x is a non-negative constant (time constant of the dynamical system). The dynamic system given by Eq. (2.11) represents the starting point in the analytic

derivation of the continuous-time ANN SE with steepest descent dynamic. It is possible to use ANN SE with steepest descent dynamic both in static and tracking state estimation. For the static case, the system of equations (2.11) is used to minimize the linearized error function (given by Eq. (2.4)) at each iteration step. In the tracking case, when the state is moving slowly, like in normal operation of a power system, the dynamical system will track the transition of the state in time. The response time of the dynamic system (2.11) is controlled by the constant τ_x. This constant has an important influence on the statistical efficiency of an ANN SE. This will later be demonstrated in the example in this Chapter.

The time evolution of the dynamic system (2.11) can be described by taking the time derivative of the error function $J(\mathbf{\Delta X})$

$$\frac{dJ(\mathbf{\Delta X})}{dt} = \sum_{i=1}^{n} \frac{\partial J(\mathbf{\Delta X})}{\partial(\Delta x_i)} \frac{d(\Delta x_i)}{dt} = -\tau_x \sum_{i=1}^{n} \left(\frac{d(\Delta x_i)}{dt}\right)^2 \leq 0, \tag{2.12}$$

where the last equality follows from Eq. (2.11). It is evident from Eq. (2.12) that the dynamic system (2.11) causes the error function $J(\mathbf{\Delta X})$ to decrease in time. The value $dJ(\mathbf{\Delta X})/dt$ is strictly less than zero for all variables $\mathbf{\Delta X}$, except at the equilibrium point, where the value is equal to zero.

2.3.1 Unconstrained Formulation

In order to derive the dynamic equations for ANN SE we write the error function given by Eq. (2.4) in expanded form

$$J(\mathbf{\Delta X}) = \frac{1}{2}\left[\mathbf{\Delta Z}^T\mathbf{\Delta Z} - 2\mathbf{\Delta X}^T\mathbf{H}^T\mathbf{\Delta Z} + \mathbf{\Delta X}^T\mathbf{H}^T\mathbf{H}\mathbf{\Delta X}\right]. \tag{2.13}$$

The first term in (2.13) does not have an influence on the optimal solution, so we can formulate the following quadratic objective function with the same fixed point $\hat{\mathbf{X}}$ as in Eq. (2.13)

$$J(\mathbf{\Delta X}) = -\frac{1}{2}\mathbf{\Delta X}^T\mathbf{W}\mathbf{\Delta X} - \mathbf{\Delta X}^T\mathbf{U}, \tag{2.14}$$

where

$$\mathbf{W} = -\mathbf{H}^T\mathbf{H} \tag{2.15}$$

is a weight matrix, and

$$\mathbf{U} = \mathbf{H}^T\Delta\mathbf{Z} \tag{2.16}$$

is an input vector.

Now the gradient of $J(\Delta\mathbf{X})$, denoted by $\nabla_x J(\Delta\mathbf{X})$, is obtained from equation (2.14)

$$\nabla_x J(\Delta\mathbf{X}) = -\mathbf{W}\Delta\mathbf{X} - \mathbf{U}. \tag{2.17}$$

Note that the above gradient of the error function (2.14) is obtained if the condition $\mathbf{W} = \mathbf{W}^T$ is fulfilled. This is always true because the matrix $\mathbf{H}^T\mathbf{H}$ is always symmetrical.

The ANN model can be derived based on Eqs. (2.11) and (2.17)

$$\tau_x \frac{d(\Delta x_i)}{dt} = \sum_{j=1}^{n} W_{ij}\Delta x_j + U_i, \quad i = 1 \ldots n, \tag{2.18}$$

where

$$W_{ij} = -\sum_{r=1}^{M} H_{ri}H_{rj}, \tag{2.19}$$

$$U_i = \sum_{r=1}^{M} H_{ri}\Delta z_r. \tag{2.20}$$

The ANN model for solving the state estimation problem is a single-layer interconnected (the number of connections depends on the $\mathbf{W}$ matrix structure) neural network with state feed-back, as shown in Figure 2.1.

Another formulation of an ANN SE is possible. If we use $\Delta\mathbf{X} = \mathbf{X}_{k+1} - \mathbf{X}_k$ to formulate the gradient of the linearized error function (2.4) as

$$\nabla_x J(\Delta\mathbf{X}) = -\mathbf{H}^T\mathbf{H}\mathbf{X}_{k+1} + \mathbf{H}^T\left[\Delta\mathbf{Z} + \mathbf{H}\mathbf{X}_k\right], \tag{2.21}$$

and since it is evident that $d(\Delta\mathbf{X})/dt = d\mathbf{X}/dt$, then from equation (2.11) follows

$$\tau_x \frac{dx_i}{dt} = \sum_{j=1}^{n} W_{ij}x_j + U_i, \qquad i = 1, \ldots n, \tag{2.22}$$

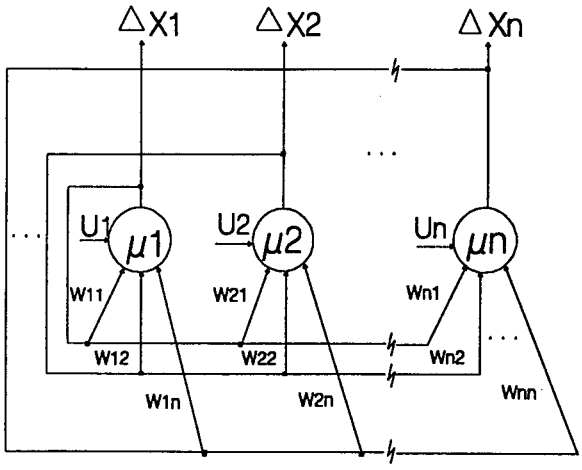

Figure 2.1: *ANN with state feed-back* ($\mu = 1/\tau$)

where

$$W_{ij} = -\sum_{r=1}^{M} H_{ri} H_{rj}, \tag{2.23}$$

$$U_i = \sum_{r=1}^{M} \left[z_r\left(k+1\right) - h_r\left(\mathbf{X}_k\right) + \sum_{j=1}^{n} H_{rj} x_j(k) \right]. \tag{2.24}$$

This form of the ANN SE directly calculates the state values and, not the state increment as in Eq. (2.18), for each iteration of the static estimation, or for each sample of measurements for the tracking estimation. Further on we are going to use only the incremental form (2.18). The results obtained in this thesis can be applied directly to the non-incremental form (2.22).

The example has been devised to demonstrate the filtering performance of the unconstrained ANN SE based on steepest descent dynamic (Eq. (2.18)). The 3-bus power system is used in this study. All the power system detail is given in Appendix D. The linear DC measurement model is used, where four active power measurements are the observations available for calculation of two system states. These states are phase angles of voltages at buses 1 and 2, denoted by x_1 and x_2.

The angle at bus 3 is the reference. The zero injection measurement at bus 1 is not taken into consideration at this stage.

The ANN SE (described by Eq. (2.18)) is tested for the situation when measurements are constant signals corrupted by small Gaussian errors. The standard deviation of the measurement errors is $0.1pu$. The starting values of two states are randomly chosen. The simulation results are presented in the fixed contour plot of the quadratic performance surface

$$\mathsf{E}\left[J\left(\mathbf{X}\right)\right] = 5.5x_1^2 + 3x_2^2 + x_1x_2 + 3.375x_1 + 4x_2 + 1.672. \tag{2.25}$$

Descent to the minimum of the mean square error function $\mathsf{E}[J\left(\mathbf{X}\right)]$ is shown in Figures 2.2 and 2.3. For $\tau_x = 1/0.005$ the ANN SE operates slower than for $\tau_x = 1/0.1$, and more samples of measurements are averaged to get one point at the performance surface. From Figures 2.2 and 2.3 we can conclude that by using the larger time constant better accuracy is achieved.

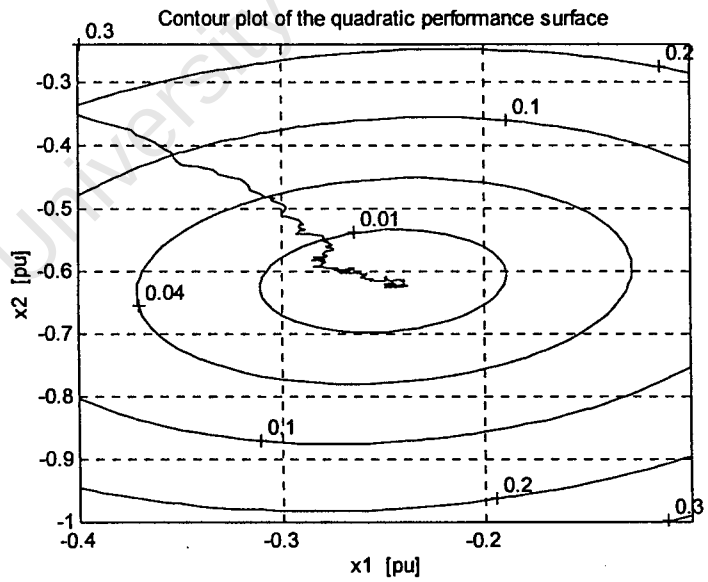

Figure 2.2: *The ANN SE simulation result for* $\tau_x = 1/0.005$

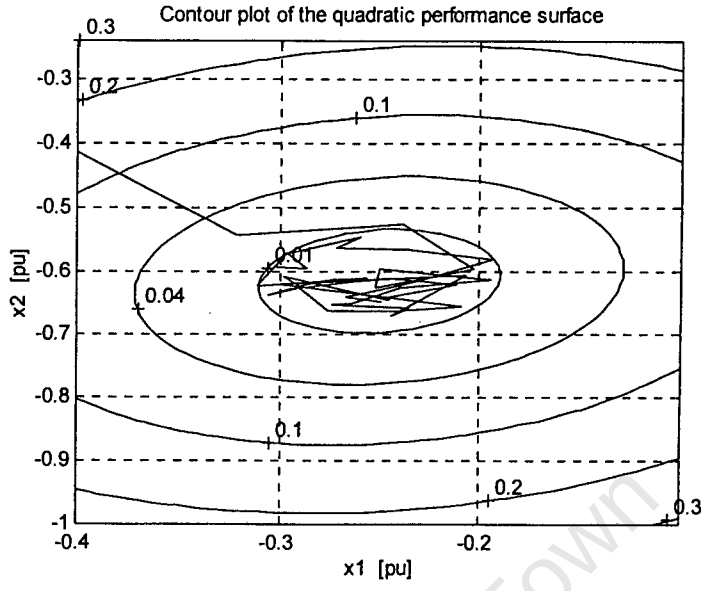

Figure 2.3: *The ANN SE simulation result for* $\tau_x = 1/0.1$

2.3.2 Constrained Formulation

The zero injections at switching stations are treated as virtual measurements. Such measurements are incorporated in the state estimation algorithm as equality constraints [9]

$$\mathbf{g}(\mathbf{X}) = \mathbf{0}. \tag{2.26}$$

The common method of solving a constrained estimation problem is by using Lagrangian multipliers [69]. For the ANN SE, a variable-reduction technique [67],[69] has been investigated. This method transforms a constrained problem into an unconstrained one, and reduces the number of independent variables. At the same time the method reduces the complexity of an ANN by reducing the number of connections [77].

Let us use the following partition of the linearized constraint matrix equation (2.26)

$$\begin{pmatrix} \mathbf{F} & \mathbf{G} \end{pmatrix} \begin{pmatrix} \Delta\mathbf{X_F} \\ \Delta\mathbf{X_G} \end{pmatrix} = 0, \tag{2.27}$$

where $\mathbf{F}$ and $\mathbf{G}$ are submatrices of the constraint equation (2.26) Jacobian matrix. $\mathbf{G}$ is a $p \times p$ non-singular matrix. It follows that the relation between $\Delta\mathbf{X_G}$ (the dependent vector of p variables) and $\Delta\mathbf{X_F}$ (the independent vector of $n-p$ variables) is

$$\Delta\mathbf{X_G} = -\mathbf{G}^{-1}\mathbf{F}\Delta\mathbf{X_F}. \tag{2.28}$$

For the state vector represented as

$$\Delta\mathbf{X} = \begin{pmatrix} -\mathbf{G}^{-1}\mathbf{F} \\ \mathbf{I} \end{pmatrix} \Delta\mathbf{X_F} = \mathbf{C}\Delta\mathbf{X_F}, \tag{2.29}$$

the reduced gradient of the error function (2.14) can be derived as follows

$$\nabla_x J(\Delta\mathbf{X_F}) = \mathbf{C}^T\nabla_x J(\Delta\mathbf{X}) = -\mathbf{C}^T\mathbf{W}\mathbf{C}\Delta\mathbf{X_F} - \mathbf{C}^T\mathbf{U}. \tag{2.30}$$

The reduced weight matrix $\tilde{\mathbf{W}} = \mathbf{C}^T\mathbf{W}\mathbf{C}$, and the input vector $\tilde{\mathbf{U}} = \mathbf{C}^T\mathbf{U}$ can be directly mapped in ANN (given by Eq. (2.18)) as:

$$\tau_x \frac{d(\Delta\mathbf{X_F})}{dt} = \tilde{\mathbf{W}}\Delta\mathbf{X_F} + \tilde{\mathbf{U}}. \tag{2.31}$$

The advantage of using a variable reduction approach compared to the Lagrangian multipliers consists in a simpler structure of the ANN SE (smaller number of connections) and in reduced computation time.

In the following example we compare:

- unconstrained ANN SE (Eq. (2.22)) in tracking mode of operation,
- constrained ANN SE (Eq. (2.31)) in tracking mode of operation,

- unconstrained WLS tracking estimator [78], and
- constrained WLS tracking estimator [9].

The 3-bus power system (all details are given in Appendix D) is represented with DC model and simulated over a period of 70 sec. The initial state of the system is given in Appendix D. We chose the following simulation scenario: Load and generation constant in the first 20 sec. After that, a sudden drop of load (from $1pu$ to $0.8pu$) took place followed by an additional slow decrease of the load in next 40 sec (from $0.8pu$ to $0.7pu$). In the last 10 sec the load and generation of the system are constant again.

The power system is observed by four active power measurements, disposed in a way shown in Appendix D. Measurement errors are simulated as normally distributed random numbers with a standard deviation equal to $0.1pu$.The zero injection at bus 1 is used as the virtual measurement in the constrained estimators. The phase angles at nodes 1 and 2 (x_1 and x_2) are estimated over 70 sec period. The time constant for unconstrained and constrained ANN estimators is $\tau_x = 1/0.05$. The interval between two measurement samples is equal to 1 sec.

Tracking trajectories of the state variable x_1, obtained for all four cases, are presented in Figure 2.4. The results for tracking WLS estimators are presented as discrete lines because they are obtained for each measurement sample at discrete moments in time. The ANN estimators continuously calculate the states for discrete inputs, so the results are presented with continuous lines. The tracking trajectories presented in Figure 2.4 clearly indicate the sensitivity of each estimator to Gaussian errors. The most insensitive to noise is the constrained ANN SE; followed in order by: unconstrained ANN SE, constrained WLS tracking, and unconstrained WLS tracking.

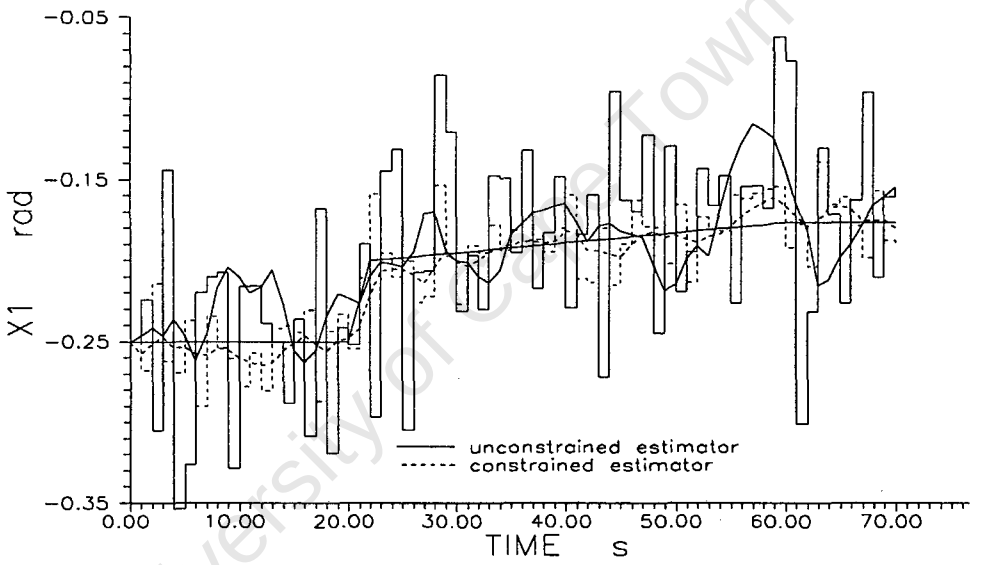

Figure 2.4: *Tracking of the state variable with WLS estimators (solid and dashed discrete lines), and ANN estimators (solid and dashed continuous lines)*

2.3.3 Algebraic Transformation of the Objective Function

In this section we use algebraic transformation of the objective function $J(\Delta\mathbf{X})$ to simplify the ANN SE implementation. The transformation preserves the characteristics of the original objective function, as shown in [68]. The transformation introduces new neurons which force the ANN to seek a saddle point rather than a minimum [68]. The transformation is

$$J(\Delta\mathbf{X}) = \frac{1}{2}(\Delta\mathbf{Z} - \mathbf{H}\Delta\mathbf{X})^T(\Delta\mathbf{Z} - \mathbf{H}\Delta\mathbf{X}) \longrightarrow$$

$$J_T(\Delta\mathbf{X}, \boldsymbol{\omega}) = (\Delta\mathbf{Z} - \mathbf{H}\Delta\mathbf{X})^T\boldsymbol{\omega} - \frac{1}{2}\boldsymbol{\omega}^T\boldsymbol{\omega}, \tag{2.32}$$

where $\boldsymbol{\omega}$ is the vector (with the same dimension as $\Delta\mathbf{Z}$) of activations of so called reversed neurons [68]. They are labelled as "reversed" because they cause the error function to increase, while the $\Delta\mathbf{X}$ neurons cause it to decrease. In obtaining the ANN SE we employ the steepest descent dynamic model (Eq. (2.11))

$$\tau_x\frac{d\Delta\mathbf{X}}{dt} = -\nabla_x J_T(\Delta\mathbf{X}, \boldsymbol{\omega}) = \mathbf{H}^T\boldsymbol{\omega}, \tag{2.33}$$

$$\tau_\omega\frac{d\boldsymbol{\omega}}{dt} = \nabla_\omega J_T(\Delta\mathbf{X}, \boldsymbol{\omega}) = -\boldsymbol{\omega} + (\Delta\mathbf{Z} - \mathbf{H}\Delta\mathbf{X}). \tag{2.34}$$

We can write the equations (2.33) and (2.34) in an augmented form as

$$\frac{d\Delta\bar{\mathbf{X}}}{dt} = \mathbf{W}\Delta\bar{\mathbf{X}} + \mathbf{U}, \tag{2.35}$$

where $\Delta\bar{\mathbf{X}}$ is given as

$$\Delta\bar{\mathbf{X}} = \begin{pmatrix} \Delta\mathbf{X} \\ \boldsymbol{\omega} \end{pmatrix}, \tag{2.36}$$

weight matrix is given as

$$\mathbf{W} = \begin{pmatrix} \mathbf{0} & \frac{1}{\tau_x}\mathbf{H}^T \\ -\frac{1}{\tau_\omega}\mathbf{H} & -\frac{1}{\tau_\omega}\mathbf{I} \end{pmatrix}, \tag{2.37}$$

and input vector is given as

$$\mathbf{U} = \begin{pmatrix} \mathbf{0} \\ \frac{1}{\tau_\omega}\mathbf{\Delta Z} \end{pmatrix}. \tag{2.38}$$

The approaching along a trajectory to a fixed-point is guaranteed if the minimizing neurons (described by Eq. (2.33)) operate at a slower rate than the maximizing neurons (described by Eq. (2.34)), i.e. $\tau_x > \tau_\omega$ [68]. A great advantage of this transformation is that the computation of $\mathbf{H}^T\mathbf{H}$ is avoided. When the power system structure changes, the changing of ANN SE weights is straightforward, based on Eq. (2.37). Another advantage is that the input vector $\mathbf{U}$ is directly related to $\mathbf{\Delta Z}$, not like in Eq. (2.18), where $\mathbf{U} = \mathbf{H}^T\mathbf{\Delta Z}$. This method is similar to Hachtel's method (Sparse Tableau Method) [79].

In the following example we compare the augmented version of the ANN SE (given by Eq. (2.35)) with the non-augmented one (given by Eq. (2.18)). Two versions are simulated on the same 3-bus test system with four active power measurements used in Section 2.3.1. The important representative results are summarized in Figures 2.5 and 2.6. Figure 2.5 shows that the reversed neurons (described by Eq. (2.34)) used in the augmented ANN SE (method 1 in the figure) result in slower but smoother convergence compared to the non-augmented ANN SE (method 2 in the figure). The result is confirmed in Figure 2.6. The augmented ANN SE (Eq. (2.35)) shows smoother performance in the presence of Gaussian measurement errors (better filtering); this is easy to see when comparing Figures 2.6 and 2.3.

2.3.4 The Hyperbolic-Tangent Activation Function

Instead of a linear activation function used in the ANN SE techniques developed till now, we can use the hyperbolic-tangent function of the following form

$$f(u) = \frac{1}{2}\left(L_u - L_l\right)\left(1 + \tanh\left(cu\right)\right) + L_l, \tag{2.39}$$

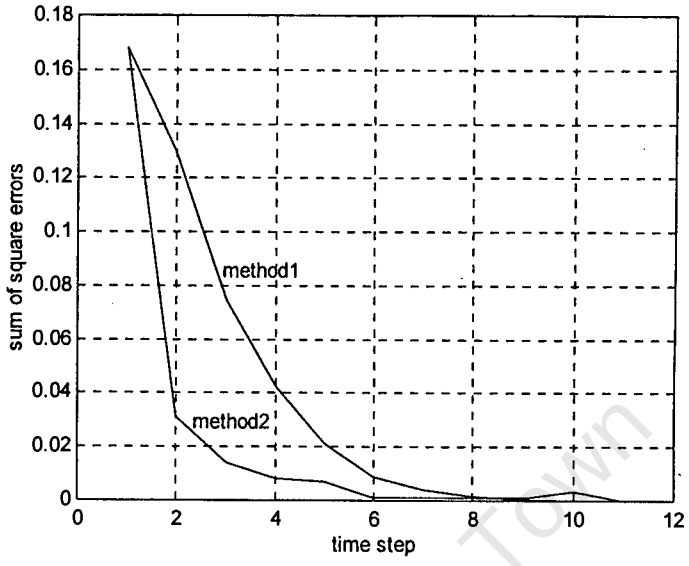

Figure 2.5: *Comparison between the augmented method (method 1) and the non-augmented method (method 2) for* $\tau_x = 1/0.1$ *and* $\tau_\omega = 1$

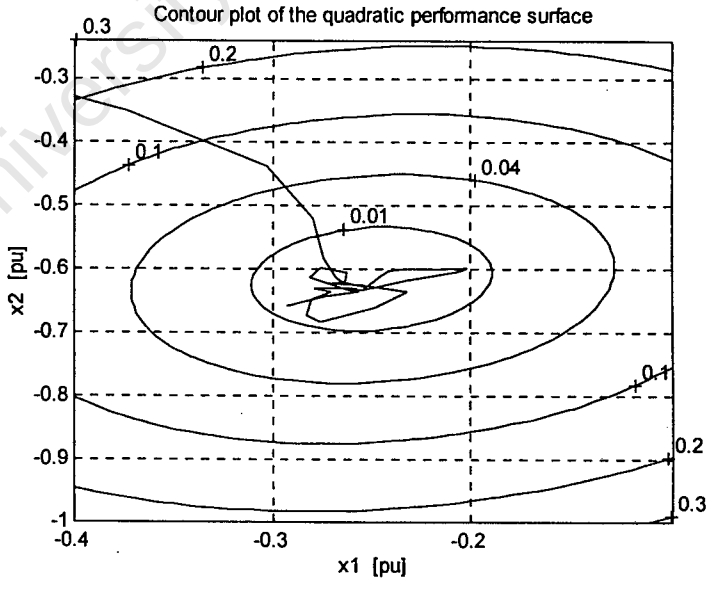

Figure 2.6: *The augmented ANN SE simulation result for* $\tau_x = 1/0.1$ *and* $\tau_\omega = 1$

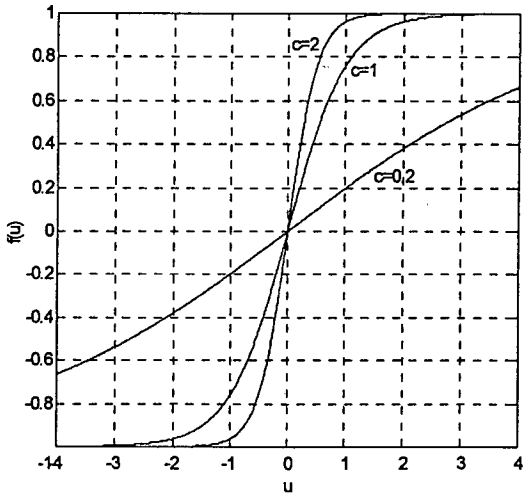

Figure 2.7: *Hyperbolic-Tangent Function $f(u)$; c - slope constant*

where L_u is the upper and L_l is the lower limit, so $f(u)$ is bounded between $L_l \leq f(u) \leq L_u$. c is a constant that defines the slope of the function. Note that as c approaches infinity, the function approaches a hard-limiting nonlinearity [45]. c is often set equal to 1. This function is shown in Figure 2.7 for $L_u = -L_l = 1$ and three different values for c. Intuitively, one can say that the saturation of the hyperbolic-tangent function, applied as an activation function in ANN SE, can contribute to suppressing random fluctuations of the state caused by measurement errors. Apart from smoothing the state transition, the advantage of this function is that it can be differentiated. This property allows a gradient search learning algorithm to be derived, as discussed later in this Chapter.

Now, we can use Eq. (2.39) as an activation function, and write the ANN SE (given by Eq. (2.18)) in the form

$$\tau_x \frac{d(\Delta x_i)}{dt} = f_i\left(\sum_{j=1}^{n} W_{ij}\Delta x_j + U_i\right), \quad i = 1 \ldots n. \tag{2.40}$$

In the next section the filtering performance of the ANN SE with the hyperbolic-tangent activation function will be analyzed.

2.3.5 Sensitivity Analysis

The estimation error, arising from input perturbations of an ANN State Estimator (given by Eq. (2.40)) in tracking mode of operation, varies depending on the set of weights $\mathbf{W}$, parameters of the activation function (Eq. (2.39)) (L_u, L_l, and c), and the response time constant τ_x. The input perturbations are measurement errors modelled as the white noise sequence. The statistical sensitivity vector of the state estimation errors to the input perturbations in an ANN SE is defined as [80]

$$\mathbf{S} \equiv \lim_{\sigma \to 0} \frac{\sqrt{\text{var}\left[\delta \mathbf{X}\right]}}{\sigma}, \tag{2.41}$$

where $\delta \mathbf{X}$ is the vector of estimation errors, σ is the standard deviation of each component of the input perturbations (measurement errors), and $\text{var}\left[\delta \mathbf{X}\right]$ is the variance of $\delta \mathbf{X}$.

Based on the definition (2.41), the formula for the sensitivity vector is derived in Appendix B as follows (for weights $\mathbf{W}$, time constant τ_x, slope constant c, and limit $L = L_u = -L_l$)

$$\mathbf{S} = \sqrt{\text{diag}\left[\left(\mathbf{I} - \exp\left(\frac{2cL\Delta t}{\tau_x}\mathbf{W}\right)\right)^{-1}\left(\exp\left(\frac{cL\Delta t}{\tau_x}\mathbf{W}\right) - \mathbf{I}\right)^2 \mathbf{W}^{-2}\mathbf{H}^T\mathbf{H}\right]}, \tag{2.42}$$

where Δt is the measurement sampling interval. The sensitivity measure of an ANN SE is derived as an average of the elements of the sensitivity vector

$$\bar{S} = \frac{1}{N}\sum_{i=1}^{N} S_i. \tag{2.43}$$

If we simplify (2.42) by introducing approximation $\exp\left(\mathbf{F}\Delta t\right) \to \mathbf{I} + \mathbf{F}\Delta t$ for small Δt, and if we fix the weights $\mathbf{W} = -\mathbf{H}^T\mathbf{H}$, a very simple approximate sensitivity index is obtained from (2.43):

$$\bar{S} \approx \sqrt{\frac{\Delta t c L}{2\tau_x}}. \tag{2.44}$$

Note that the sensitivity index $\bar{S}$ does not depend on $\mathbf{W}$ when $\mathbf{W} = -\mathbf{H}^T\mathbf{H}$. In this case the ANN SE is designed using an analytic approach, i.e. the weights are fixed values obtained directly from the measurement model. However, if the ANN SE is designed by using the synthetic approach, i.e. weights are obtained by training, the more general formula to calculate the sensitivity index (2.43) using sensitivity vector (2.42) must be used.

If the measurement sampling interval Δt, the slope constant c or the activation function limit L decreases, implies a smaller sensitivity index $\bar{S}$, and consequently the filtering performance of an ANN SE will be improved. On the contrary, if the time constant τ_x decreases, the response of an ANN SE speeds up and this will degrade the filtering performance. A trade-off between filtering performance, speed of operation, and sampling interval is required as is obvious from formula (2.44).

In the following example the influence of the shape of the activation function on the ANN SE filtering performance (sensitivity to small measurement errors) is demonstrated. The estimator is analytically designed for the 3-bus power system with 4 active power measurements (Appendix D) as the ANN with steepest descent dynamic given by Eq. (2.40). The time evolution of the ANN SE with the hyperbolic-tangent activation function (slope constant $c = 0.1$ and limit $L = 1$) is shown in Figure 2.8. The figure shows that the ANN converges at a point close to the solution (bottom of the bowl), and then jitters around the solution due to small random errors in measurements ($\sigma = 0.1pu$). The sensitivity index for this case, calculated by using (2.43) and (2.42), is equal to $\bar{S} = 0.1816$. The simulation results are obtained for 40 samples of statistically stationary measurements whose values are given in Appendix D. The interval between two samples is $\Delta t = 1$ time unit. When we increase the slope constant from $c = 0.1$ to $c = 0.4$, the jittering around the solution increases, as shown in Figure 2.9. The filtering performance decays. The sensitivity index

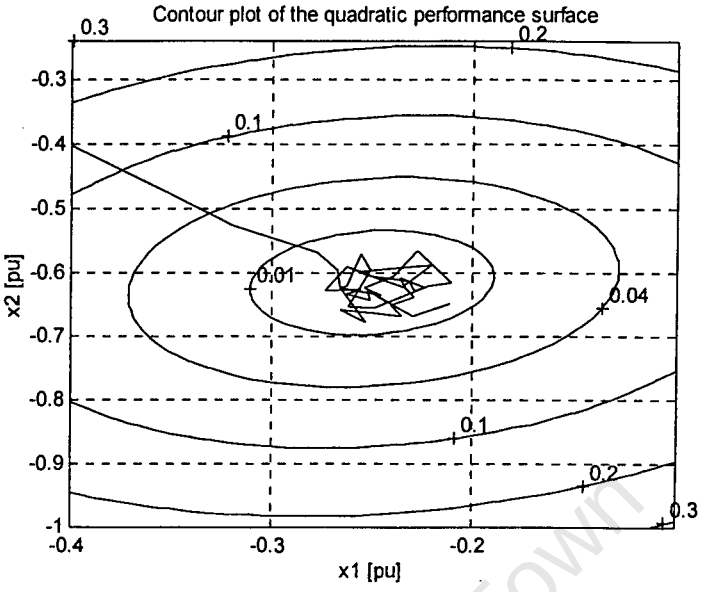

Figure 2.8: *The ANN SE simulation result for* $\tau_x = 1/0.1$ *and* $c = 0.1$

increases to $\bar{S} = 0.2153$. The ANN SE becomes more sensitive to measurement errors represented as a white sequence.

2.3.6 Real Time Recurrent Learning

The Real Time Recurrent Learning (RTRL) algorithm presented in this section is continuous and a recurrent generalization of a widely used back propagation algorithm [45][49]. It represents the basis for synthetic design of an ANN SE with steepest descent dynamic.

The learning method is derived using the gradient search technique for minimizing a sum of weighted squares of measurement residuals. The criterion function is given by

$$E(\mathbf{W}) = \frac{1}{2}\sum_{i=1}^{M} p_i r_i(\mathbf{W})^2, \tag{2.45}$$

where

$$r_i(\mathbf{W}) = z_i - h_i\left(\hat{\mathbf{X}}(\mathbf{W})\right) \tag{2.46}$$

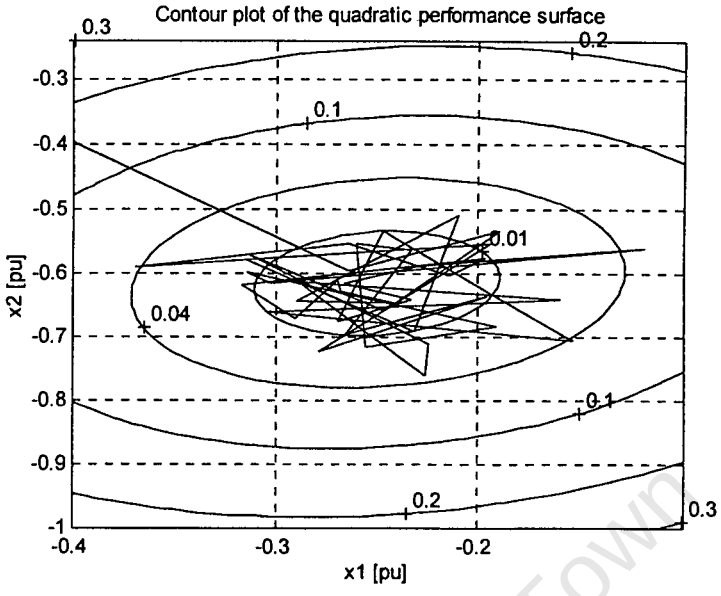

Figure 2.9: *The ANN SE simulation result for* $\tau_x = 1/0.1$ *and* $c = 0.4$

is the residual of $i-th$ measurement, and p_i is the weighting parameter. $\hat{\mathbf{X}}(\mathbf{W})$ is the state estimate; i.e. the output of an ANN SE with connection weights $\mathbf{W}$.

The minimization of $E(\mathbf{W})$ is controlled by the steepest descent dynamic

$$\tau_w \frac{d\mathbf{W}}{dt} = -\nabla_w E(\mathbf{W}) \tag{2.47}$$

where τ_w is a non-negative constant, which defines the time scale over which the weight matrix $\mathbf{W}$ changes. Based on dynamic equations (2.47) the system with ANN structure, which adjusts the weight matrix $\mathbf{W}$ in ANN SE with the steepest descent dynamic (2.40), is derived in Appendix C. The complete ANN SE dynamic system consists of three sets of differential equations, summarized as follows

$$\tau_x \frac{d\Delta x_i}{dt} = f_i\left(\sum_{j=1}^{n} W_{ij}\Delta x_j + U_i\right), \qquad i = 1\ldots n, \tag{2.48}$$

$$\tau_y \frac{dy_i}{dt} = \sum_{j=1}^{n} W_{ji} y_j + T_i, \qquad i = 1\ldots n, \tag{2.49}$$

$$\tau_w \frac{dW_{ij}}{dt} = y_i \Delta x_j, \qquad i,j = 1\ldots n. \tag{2.50}$$

where $T_i = \sum_{j=1}^{M} p_j \frac{\partial h_j}{\partial x_i} r_j$ and y_i are the $i-th$ input and output respectively, of the dynamical system (2.49). The formula for calculating T_i is derived in Appendix C.

The ANN SE together with RTRL evolve at the same time in the space of activations (state space) and in the space of weights (parameter space). The evolution in the state space is determined by Eq. (2.48). These equations are called forward propagation equations. The evolution in the parameter space is determined by Eqs. (2.49) and (2.50), so called backward propagation equations [49].

For the ANN SE to learn, it is necessary that the time constants are properly chosen. The time constant of the forward propagation equation (2.48) is τ_x. The time constant of the backward propagation equation (2.49) is τ_y and the adaptation time constant of the system (2.50) is τ_w. It is straightforward to establish relationships which must be satisfied by the time constants of the ANN SE. First, note that $y_i \Delta x_j$ is the correct form of the gradient estimate in Eq. (2.50) only if $\mathbf{\Delta X}$ and $\mathbf{Y}$ are the steady state solutions of equations (2.48) and (2.49). This observation will help in defining the relationship between time constants τ_x, τ_y, and τ_w. Let τ_s be the time constant over which the input fluctuates (determined by the interval between two sample measurements). Then for the dynamic systems (2.48) and (2.49) to operate near steady state it is necessary that the solution of both systems relax with a characteristic time faster than τ_s, i.e. $\tau_s > \tau_x$ and $\tau_s > \tau_y$. Also, for dynamic systems (2.48) and (2.49) to operate near steady state it is necessary for $\mathbf{W}$ to change slowly relative to the relaxation times of $\mathbf{\Delta X}$ and $\mathbf{Y}$; thus one must also have $\tau_w > \tau_x$ and $\tau_w > \tau_y$. We also set the condition that $\mathbf{Y}$ should change slower than $\mathbf{\Delta X}$, i.e. $\tau_y > \tau_x$. Finally, if we want to keep the weights from just tracking the fluctuations, it is necessary that $\tau_w > \tau_s$. Those relationships imply that time constants are ordered in the following way

$$\tau_w > \tau_s > \tau_y > \tau_x. \tag{2.51}$$

The above relationship between time constants affects on the capability of the ANN SE to suppress the influence of small errors on the learning process, and to keep the behavior of the ANN SE stable.

In the following example the learning process of the ANN SE with steepest descent dynamic (2.48) is demonstrated. Figure 2.10 presents the minimization process of the sum of square errors function (2.45) during the ANN SE training. The ANN SE is trained for the 3-bus power system observed by true values of 4 active power measurements (specified in Appendix D). The parameters for the estimator are chosen as $\tau_x = 1/0.1$ and $c = 0.2$. We used as starting weights of the ANN SE values obtained from a random number generator. The weights are adjusted using dynamical systems (2.49) and (2.50) with time constants $\tau_y = 1/0.01$ and $\tau_w = 1/0.001$. The evolution of the complete dynamical system (estimator + learning) in the state space ($\mathbf{X}$) is shown in Figures 2.11 and 2.12. The evolution in the space of weights ($\mathbf{W}$) is shown in Figure 2.13. The final values obtained for the weight matrix are

$$\mathbf{W}_1 = -\begin{pmatrix} 3.4362 & 1.2685 \\ 1.2685 & 2.6308 \end{pmatrix}.$$

The performance of the synthesized ANN SE is tested on the same power system, but now all of the existing four active power measurements are corrupted with normally distributed random errors with $\sigma = 0.1pu$. The results are obtained for 40 samples of measurements and shown on the contour plot of the mean error function $\mathsf{E}[J(\mathbf{X})]$ (Figure 2.14), where $\mathsf{E}[J(\mathbf{X})]$ is the function (2.25). The interval between two samples is $\Delta t = 1$ time unit. Note that if we repeat the learning process with the same parameters but different starting weights (randomly obtained) the result is completely different

$$\mathbf{W}_2 = -\begin{pmatrix} 12.2357 & -2.0111 \\ -2.0111 & 3.9426 \end{pmatrix}.$$

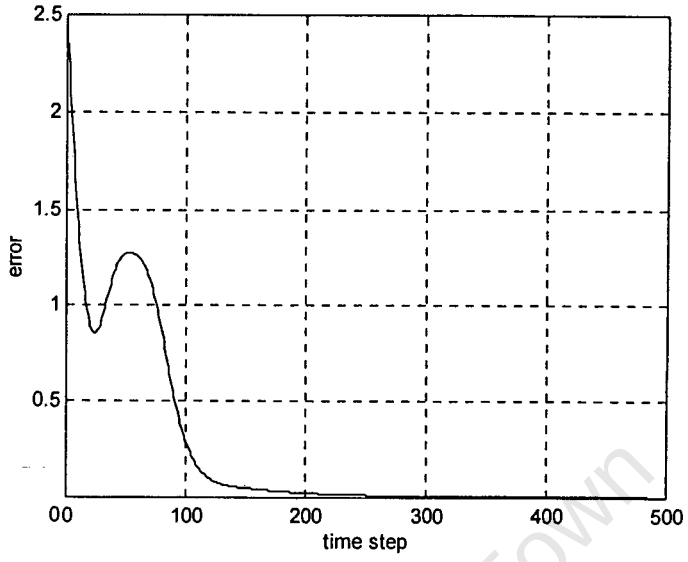

Figure 2.10: *Time evolution of the ANN SE learning error function*

The filtering performance of the ANN SE with new set of weights is improved, as shown in Figure 2.15. For the first set of weights $\mathbf{W}_1$ the sensitivity measure is equal to $\bar{S} = 0.2231$ and for the second one $\mathbf{W}_2$ equal to $\bar{S} = 0.1562$. This confirms the results obtained in simulation. The sensitivity of the ANN SE output on white sequence measurement errors depends on the ANN SE weights.

2.4 ANN State Estimator with Hopfield-style Dynamic

2.4.1 Formulation of the Method

In order to develop the Hopfield ANN for state estimation, the quadratic error function (2.14) is expanded into [45]

$$J_H(\Delta\mathbf{X}) = -\frac{1}{2}\Delta\mathbf{X}^T\mathbf{W}\Delta\mathbf{X} - \Delta\mathbf{X}^T\mathbf{U} + \sum_{i=1}^{n}\int_0^{\Delta x_i} f^{-1}(\alpha)\,d\alpha, \tag{2.52}$$

where $f(\cdot)$ is the hyperbolic-tangent function and it is defined in the expression (2.39). The elements of the weight matrix $\mathbf{W}$ will not correspond directly to $\mathbf{H}$,

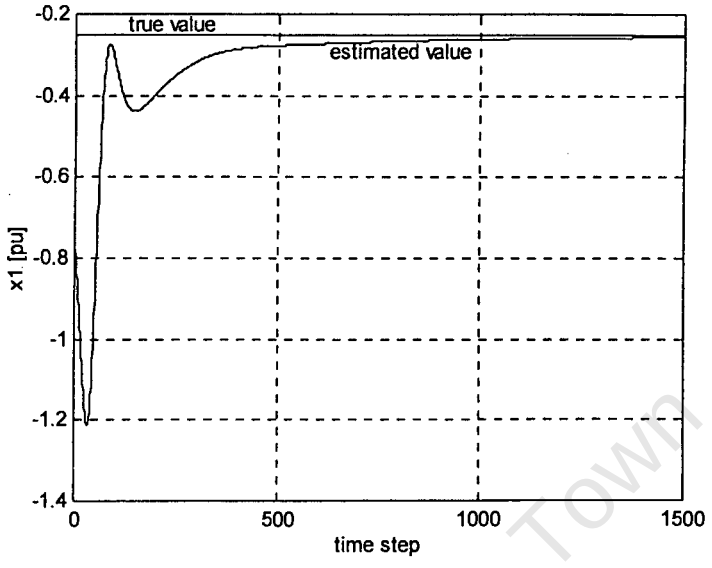

Figure 2.11: *Time evolution of the state x_1 during ANN SE learning*

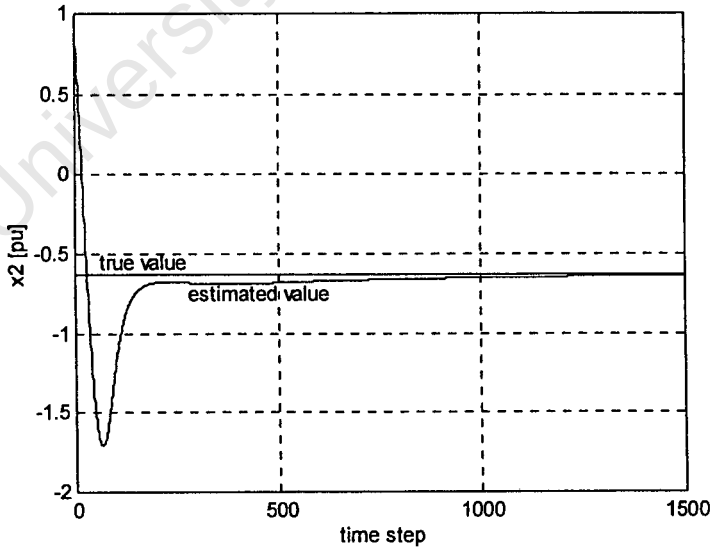

Figure 2.12: *Time evolution of the state x_2 during ANN SE learning*

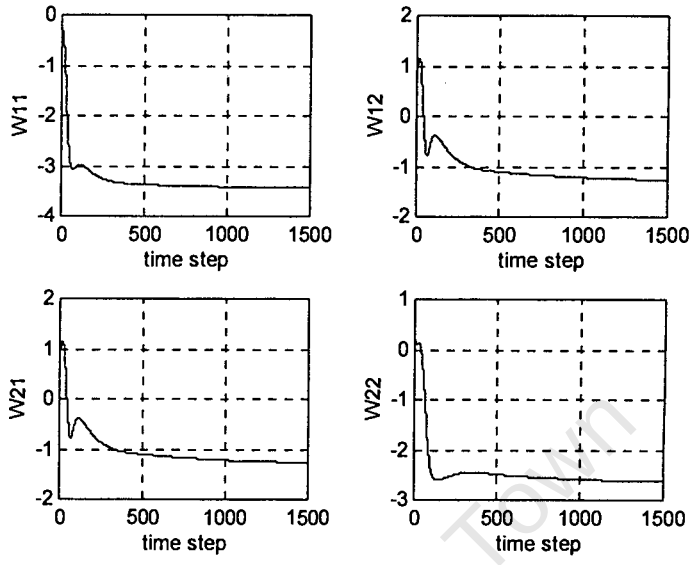

Figure 2.13: *Time evolution of the weights during the ANN SE learning*

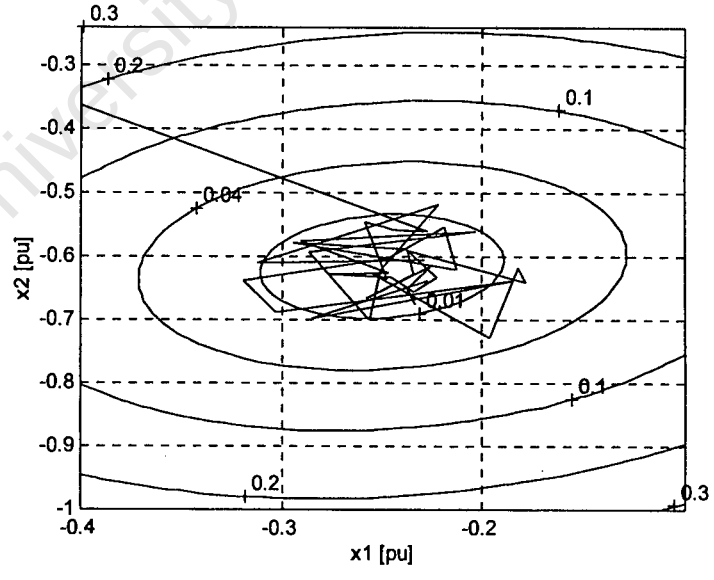

Figure 2.14: *The ANN SE simulation result for* $\tau_x = 1/0.1$, $c = 0.2$, *and* $\mathbf{W}_1$

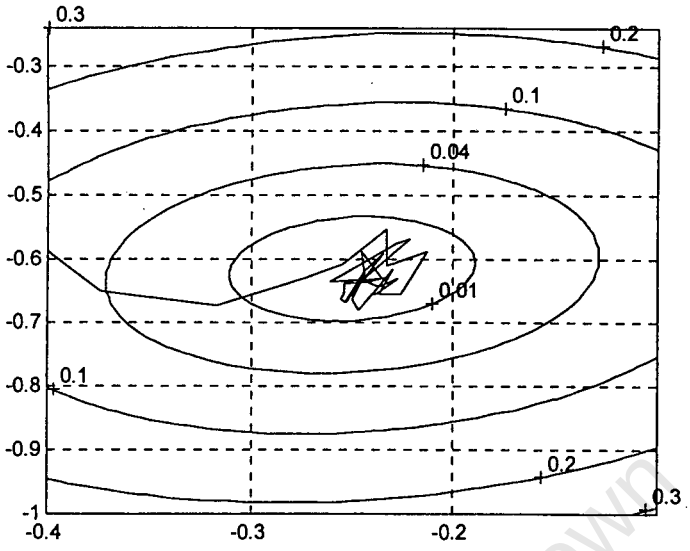

Figure 2.15: *The ANN SE simulation result for* $\tau_x = 1/0.1$, $c = 0.2$, *and* $\mathbf{W}_2$

as in Eq. (2.14). The weights will have to be determined by using an appropriate learning technique.

The Hopfield ANN SE can be obtained by applying the following dynamic ANN system for minimizing $J_H(\mathbf{\Delta X})$ [45]

$$\tau_x \frac{d\mathbf{u}}{dt} = -\nabla_x J_H(\mathbf{\Delta X}), \tag{2.53}$$

$$\mathbf{\Delta X} = \mathbf{f}(\mathbf{u}).$$

The evolution in time of the dynamic ANN system (2.53) can, similarly to Eq. (2.12), be described by taking the time derivative of the error function $J_H(\mathbf{\Delta X})$

$$\frac{dJ_H(\mathbf{\Delta X})}{dt} = \sum_{i=1}^{n} \frac{\partial J_H(\mathbf{\Delta X})}{\partial(\Delta x_i)} \frac{d(\Delta x_i)}{dt} = -\tau_x \sum_{i=1}^{n} f'(u_i) \left(\frac{du_i}{dt}\right)^2, \tag{2.54}$$

where the last equality follows from equation (2.53). Note that the condition $\mathbf{W}^T = \mathbf{W}$ have to be fulfilled [45]. The derivative of the hyperbolic-tangent function (2.39) is

$$f'(u) = \frac{c}{2}(L_u - L_l)\left(1 - \tanh^2(cu)\right) \geq 0. \tag{2.55}$$

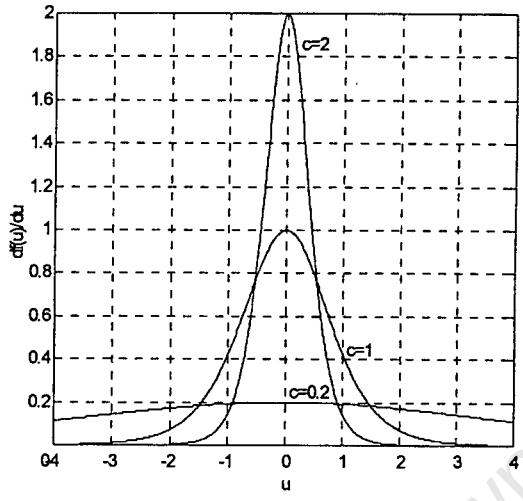

Figure 2.16: *Derivative of the hyperbolic-tangent function $f(u)$; c - slope constant*

The function is shown in Figure 2.16 for $L_u = 1$ and $L_l = -1$. Hence, we obtained that $dJ_H(\mathbf{\Delta X})/dt \leq 0$. Thus, if $\mathbf{W}$ is symmetric, the dynamic system (2.53) will reach a fixed equilibrium point.

The Hopfield network can be derived according to Eqs. (2.53) and (2.52)

$$\tau_x \frac{du_i}{dt} = -u_i + \sum_{j=1}^{n} W_{ij} f_j(u_j) + U_i, \quad i = 1 \ldots n. \tag{2.56}$$

Rather than using the above form of the Hopfield network it has been found convenient to transform these equations into the form

$$\tau_x \frac{d\Delta x_i}{dt} = -\Delta x_i + f_i\left(\sum_{j=1}^{n} W_{ij}\Delta x_j + U_i\right), \quad i = 1 \ldots n. \tag{2.57}$$

This is a Continuous Time Recurrent Neural Network (CTRNN) [45][49]. The structure of a CTRNN is similar to the network shown in Figure 2.1. The transformation from the Hopfield network (2.56) into the CTRNN is done by a simple transformation $\mathbf{u} = \mathbf{W\Delta X} + \mathbf{U}$ [49]. In the CTRNN model, used as an ANN SE, we can not directly apply the known power network parameters in calculating the CTRNN parameters. The analytical approach is not possible. The synthetic approach is

the only possible way of designing the ANN SE for specific estimation problems. A CTRNN parameters could be determined by using the Real Time Recurrent Learning (RTRL) procedure [49]. The RTRL method for a CTRNN estimator is presented further on in this section.

2.4.2 Sensitivity Analysis

Many different sets of connection weights and parameters of the activation function can realize input-output mapping in an ANN SE with a Hopfield-style dynamic used in tracking estimation. The formula for calculation of the sensitivity index, which gives the statistical sensitivity of the estimation error versus the measurement errors (input perturbations) is

$$\bar{S} = \frac{1}{N}\sum_{i=1}^{N} S_i \tag{2.58}$$

where S_i is the $i-th$ component of the sensitivity vector. Based on definition (2.41) the approximate formula for calculating the sensitivity vector is derived in Appendix B

$$\mathbf{S} = \sqrt{\frac{\Delta t\,(cL)^2}{2\tau_x}\,\mathrm{diag}\left[(\mathbf{I}-cL\mathbf{W})^{-1}\,\mathbf{H}^T\mathbf{H}\right]}. \tag{2.59}$$

where Δt is the sampling interval. The approximative formula (2.59) is derived by assuming that for the measurement errors the covariance matrix $\mathbf{R} = \sigma^2\mathbf{I}$ applies.

From formula (2.59) we can conclude that the filtering performance of an ANN SE with Hopfield-style dynamic will improve if we reduce the sampling interval Δt, the slope constant c or the activation function limit L. Conversely, reducing the time constant τ_x will speed up the response of an ANN SE, and will degrade the filtering performance. The sensitivity index also depends on weights obtained after learning.

2.4.3 Real Time Recurrent Learning

The Real Time Recurrent Learning (RTRL) method provides the basis for the synthetic design of an ANN SE with Hopfield-style dynamic (CTRNN). In the same way as in Paragraph 2.3.6 the RTRL method for CTRNN is derived in Appendix C using the ANN steepest descent dynamic system (2.47) for minimization of the objective function (2.45). The complete dynamic system for an ANN SE with Hopfield-style dynamic (2.57) and including RTRL is

$$\tau_x \frac{d\Delta x_i}{dt} = -\Delta x_i + f_i\left(\sum_{j=1}^{n} W_{ij}\Delta x_j + U_i\right), \qquad i = 1 \ldots n, \tag{2.60}$$

$$\tau_y \frac{dy_i}{dt} = -y_i + f_i'\left(\sum_{j=1}^{n} W_{ij}\Delta x_j + U_i\right)\left(\sum_{j=1}^{n} W_{ji} y_j + T_i\right), \qquad i = 1 \ldots n, \tag{2.61}$$

$$\tau_w \frac{dW_{ij}}{dt} = y_i \Delta x_j, \qquad i, j = 1 \ldots n. \tag{2.62}$$

As for the steepest descent version presented in Paragraph 2.3.6, the CTRNN, together with RTRL, evolve at the same time in the space of activations (state space) and in the space of weights (parameter space). The evolution in the state space is determined by CTRNN dynamics (2.60), and in the parameter space by RTRL dynamics (2.61) and (2.62) [49]. The time constants τ_x, τ_y, and τ_w must satisfy Eq. (2.51).

2.5 Implementation

2.5.1 Continuous-Time ANN State Estimator

A way of achieving real-time operation of the ANN SE would be to map the ANN structure and weights into hardware [51]. A short outline of hardware implementation of an ANN SE with Hopfield style dynamic is presented here [76]. The hardware implementation of steepest descent style ANN SE is very similar, and is

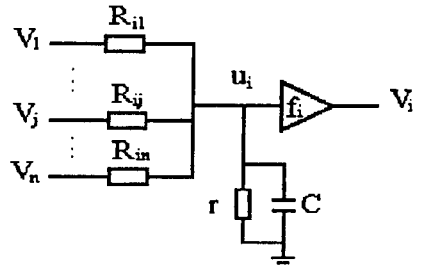

Figure 2.17: *Neuron circuit*

discussed in Ref. [52][53]. The activation unit (neuron) i, of the ANN SE is composed of the circuit shown in Figure 2.17. The symbols in Figure 2.17 have the following meanings: u_i is the activation input voltage, V_i is the output voltage, and the operational amplifier has the hyperbolic-tangent transfer function $V_i = f_i(u_i)$. The input of each unit is connected to the ground with a resistor r in parallel with capacitor C, and the output of unit j is connected to the input of unit i with a resistor R_{ij}. The equation for the neuron i is

$$C\frac{du_i}{dt} + \frac{u_i}{r} = \sum_{j=1}^{n} \frac{1}{R_{ij}} (V_j - u_i), \tag{2.63}$$

or equivalently

$$\tau_i \frac{du_i}{dt} = -u_i + \sum_{j=1}^{n} W_{ij} f_j(u_j), \tag{2.64}$$

where

$$\tau_i = R_i C, \tag{2.65}$$

$$\frac{1}{R_i} = \frac{1}{r} + \sum_{j=1}^{n} \frac{1}{R_{ij}}, \tag{2.66}$$

$$W_{ij} = \frac{R_i}{R_{ij}}. \tag{2.67}$$

If we add an external input to the activation unit i equation (2.64) is identical to Eq. (2.56). For any starting state the values of the ANN SE circuit settles down to a steady state, as proved in Paragraph 2.4.1.

This circuit has been realized in analog VLSI technology [51][76]. The implementation involves choosing the resistances R_{ij} to satisfy Eq. (2.67). Resistors could not be changed once made [76]. Another approach employed active elements (transistors) for the connections instead of passive resistive elements. Most importantly, they can be programmed after fabrication, allowing the use of general purpose chips and the possibility of chip-in loop learning [81]. Also, it is important to make the connections non-volatile, so that they may retain their values even when the device is turned off, as covered in Ref. [55].

2.5.2 Discrete-Time ANN State Estimator

Discrete-time ANN SE can be viewed as numerical discretization of continuous-time ANN SE given by Eqs. (2.40) and (2.57). The continuous-time dynamical equations can be solved with the usual numerical techniques used for integrating differential equations (Euler or Runge-Kutta). It turns out that for the ANN SE the Euler method (also called first-order finite difference) suffices for convergence to the solution [49]. For a constant integration step Δt, Euler's method transforms the system of differential equations (2.48), (2.49) and (2.50) (steepest descent dynamical system with RTRL equations) into a system of difference equations

$$\Delta x_i(q) = \Delta x_i(q-1) + \mu_x f_i\left(\sum_{j=1}^{n} W_{ij}\Delta x_j(q-1) + U_i\right), \qquad i = 1\ldots n, \tag{2.68}$$

$$y_i(q) = y_i(q-1) + \mu_y\left(\sum_{j=1}^{n} W_{ji} y_j(q-1) + T_i(q)\right), \qquad i = 1\ldots n, \tag{2.69}$$

$$W_{ij}(q) = W_{ij}(q-1) + \mu_w y_i(q)\,\Delta x_j(q), \qquad i,j = 1\ldots n. \tag{2.70}$$

where $\mu_x = \Delta t/\tau_x$, $\mu_y = \Delta t/\tau_y$ and $\mu_w = \Delta t/\tau_w$.

Similarly, for the discrete form of CTRNN, so called Discrete-Time Recurrent Neural Network (DTRNN), with RTRL (equations (2.60), (2.61) and (2.62)) the difference equations are

$$\Delta x_i(q) = \mu_x f_i\left(\sum_{j=1}^{n} W_{ij}\Delta x_j(q-1) + U_i\right), \qquad i = 1\ldots n, \tag{2.71}$$

$$y_i(q) = \mu_y f_i'\left(\sum_{j=1}^{n} W_{ij}\Delta x_j(q) + U_i\right)\left(\sum_{j=1}^{n} W_{ji} y_j(q-1) + T_i(q)\right), \qquad i = 1\ldots n, \tag{2.72}$$

$$W_{ij}(q) = W_{ij}(q-1) + \mu_w y_i(q)\,\Delta x_j(q), \qquad i,j = 1\ldots n. \tag{2.73}$$

For the solution of these difference equations to be stable, Δt must be less than τ_x.

The dynamic system (2.72) includes the derivative of the activation function, so it is necessary to find the formula for calculating its value. In the case of the hyperbolic-tangent activation function (2.39) we can write the derivative as

$$f_i' = \frac{2c}{(L_u - L_l)}\left(L_l^2 - f_i^2\right). \tag{2.74}$$

From formula (2.74) we can see that the value of the derivative of the function directly follows from the value of the function. When the upper and the lower limits of the activation function $f_i(\cdot)$ are $L_u = -L_l = 1$, the formula for calculating the derivative is further simplified to $f_i' = c\,(1 - f_i^2)$.

The recursive equations developed here are used in computer simulation studies of different ANN SE's throughout the thesis. These equations also form the basis for the digital computer implementation of ANN SE.

Chapter 3

ANN State Estimator: Feasible Methods

Fundamentals of the novel approach to state estimation based on the ANN theory have been presented in Chapter 2. Two classes of ANN SE methods have been proposed: ANN SE with steepest descent dynamic and ANN SE with Hopfield-style dynamic. It is possible to make these methods more efficient for practical implementation by using some unique properties of the power systems (Appendix A). Therefore, in this chapter these properties are used in order to create practically feasible ANN SE methods. Simplified ANN estimators are developed for static and tracking modes of implementation. New methods are tested in simulation studies. The simulation results, together with the sensitivity analysis, are reported at the end of the chapter.

3.1 Decoupled ANN State Estimator with Steepest Descent Dynamic

The starting point in derivation of Decoupled ANN SE with steepest descent dynamic is the non-linear measurement vector function (2.1) (defining the relation between measurements $\mathbf{Z}$ and states $\mathbf{X}$). Full representation of the measurement vector function $\mathbf{h}(\mathbf{X})$ and its derivatives $\mathbf{H}(\mathbf{X}) = \frac{\partial \mathbf{h}(\mathbf{X})}{\partial \mathbf{X}}$ are given in Appendix A.

The state vector $\mathbf{X}$ of a power system consists of two subvectors:

- $\mathbf{V}$ is the $(N \times 1)$ vector whose elements are the bus voltage magnitudes.
- $\mathbf{\Theta}$ is the $(N \times 1)$ vector whose elements are the bus voltage angles, including the angular reference.

The measurement vector $\mathbf{Z}$ (dimension $M = M_P + M_Q$) can be partitioned into two subvectors:

- $\mathbf{Z}_P$ is the $(M_P \times 1)$ vector which includes active power flows P_{ij}, and active power injections P_i.
- $\mathbf{Z}_Q$ is the $(M_Q \times 1)$ vector which includes reactive power flows Q_{ij}, reactive power injections Q_i and voltage magnitudes V_i.

In normalized version flows P_{ij} and Q_{ij}, and injections P_i and Q_i are replaced by P_{ij}/V_i, Q_{ij}/V_i, P_i/V_i and Q_i/V_i respectively.

The ANN SE with steepest descent dynamic, analytically developed in the Chapter 2, has state dependent weight matrix $\mathbf{W}$ and input vector $\mathbf{U}$ (equations (2.14) and (2.15)). This makes practical implementation difficult. In order to create state independent $\mathbf{W}$ and $\mathbf{U}$ values, the Jacobian matrix $\mathbf{H}$ given by Eq. (2.7) needs to be approximated with a constant matrix.

First, by using the property of weak coupling between $P-V$ and $Q-\theta$ in power systems [10] the Jacobian matrix $\mathbf{H}$ is approximated as

$$\mathbf{H} = \begin{pmatrix} \mathbf{H}_{P\theta} & 0 \\ 0 & \mathbf{H}_{QV} \end{pmatrix}, \tag{3.1}$$

where the dimensions of submatrices are $\mathbf{H}_{P\theta}$ - $(M_P \times N)$ and $\mathbf{H}_{QV}$ - $(M_Q \times N)$. Second, the additional approximations are introduced to create $\mathbf{H}_{P\theta}$ and $\mathbf{H}_{QV}$ constant (state independent) matrices. These approximations are described in Appendix A.

The linearization of the measurement model, and the application of decoupling principle allows development of two independent neural networks

$$\tau_p \frac{d(\Delta\theta_i)}{dt} = f_i\left(\sum_{j=1}^{N} W_{Pij}\Delta\theta_j + U_{Pj}\right), \quad i = 1\ldots N, \tag{3.2}$$

$$\tau_q \frac{d(\Delta V_i)}{dt} = f_i\left(\sum_{j=1}^{N} W_{Qij}\Delta V_j + U_{Qj}\right), \quad i = 1\ldots N, \tag{3.3}$$

where τ_p and τ_q are time constants, and

$$W_{Pij} = -\sum_{r=1}^{M_P} H_{P\theta ri} H_{P\theta rj}, \tag{3.4}$$

$$U_{Pi} = \sum_{r=1}^{M_P} H_{P\theta ri}\Delta z_{pr}, \tag{3.5}$$

$$W_{Qij} = -\sum_{r=1}^{M_Q} H_{QVri} H_{QVrj}, \tag{3.6}$$

$$U_{Qi} = \sum_{r=1}^{M_Q} H_{QVri}\Delta z_{qr}. \tag{3.7}$$

According to Eq. (2.5) residuals Δz_{pr} and Δz_{qr} are the elements of vectors

$$\mathbf{\Delta Z}_P = \mathbf{Z}_P - \mathbf{h}_P(\mathbf{\Theta}_k, \mathbf{V}_k), \tag{3.8}$$

and

$$\mathbf{\Delta Z}_Q = \mathbf{Z}_Q - \mathbf{h}_Q(\mathbf{\Theta}_k, \mathbf{V}_k), \tag{3.9}$$

respectively. In the above equations the non-linear measurement vector function $\mathbf{h}(\mathbf{\Theta}, \mathbf{V})$ is divided into two sets of vector functions $\mathbf{h}_P(\mathbf{\Theta}, \mathbf{V})$ and $\mathbf{h}_Q(\mathbf{\Theta}, \mathbf{V})$, according to the active and the reactive sets of measurements (see Appendix A).

3.1.1 Static State Estimation

The ANN SE given by Eqs. (3.2) and (3.3) can be incorporated in the static state estimation algorithm in the following way:

1. Initialise iteration counter $k = 0$ and set the values of $\boldsymbol{\Theta}_k$ and $\mathbf{V}_k$ (flat start).
2. Compute the transformed residual vector $\Delta\mathbf{Z}_P = \mathbf{R}^{-1/2}\left(\mathbf{Z}_P - \mathbf{h}_P\left(\boldsymbol{\Theta}_k, \mathbf{V}_k\right)\right)$.
3. Compute the input vector $U_{Pi} = \sum_{j=1}^{M_P} H_{P\theta ji}\Delta z_{pj}, \qquad i = 1 \ldots N.$
4. Compute $\Delta\boldsymbol{\Theta}$ from $\tau_p \frac{d\Delta\theta_i}{dt} = f_i\left(\sum_{j=1}^{N} W_{Pij}\Delta\theta_j + U_{Pj}\right), \quad i = 1 \ldots N$
   which minimizes ANN objective function $J\left(\Delta\boldsymbol{\Theta}\right)$ given in Eq. (2.14).
5. Compute $\boldsymbol{\Theta}_{k+1} = \boldsymbol{\Theta}_k + \Delta\boldsymbol{\Theta}$.
6. Compute the transformed residual vector $\Delta\mathbf{Z}_Q = \mathbf{R}^{-1/2}\left(\mathbf{Z}_Q - \mathbf{h}_Q\left(\boldsymbol{\Theta}_{k+1}, \mathbf{V}_k\right)\right)$.
7. Compute the input vector $U_{Qi} = \sum_{j=1}^{M_Q} H_{QVji}\Delta z_{qj}, \qquad i = 1 \ldots N.$
8. Compute $\Delta\mathbf{V}$ from $\tau_q \frac{d\Delta V_i}{dt} = f_i\left(\sum_{j=1}^{N} W_{Qij}\Delta V_j + U_{Qj}\right), \quad i = 1 \ldots N$
   which minimizes ANN objective function $J\left(\Delta\mathbf{V}\right)$ given in Eq. (2.14).
9. Compute $\mathbf{V}_{k+1} = \mathbf{V}_k + \Delta\mathbf{V}$.
10. If $\Delta\boldsymbol{\Theta}$ and $\Delta\mathbf{V}$ are smaller than threshold, then stop.
11. $k = k + 1$, go to 2.

This algorithm calculates the state of a power system by using only one sample of measurements. The performance of the algorithm is tested in the simulation study. The results and discussions are presented further on in this chapter.

3.1.2 Tracking State Estimation

Tracking state estimation is based on the assumption that the power system state vector $\mathbf{X}$ does not change or that it changes very slowly between two consecutive sample measurements, q and $q + 1$ respectively. The initial values for the tracking algorithm at the sample $q + 1$ are the estimates $\hat{\mathbf{X}}(q)$ at the previous sample q.

The application of ANN estimators given by Eqs. (3.2) and (3.3) is straightforward. For each new measurement sample we need to calculate the residuals (3.8), and then to minimize the objective function of the ANN SE represented in equations (3.2) and (3.3). The active ANN given by Eq. (3.2) and the reactive ANN given

by Eq. (3.3) can operate in sequential or in parallel mode. The sequential mode is the same as in the static case, i.e. the result of the active ANN is used immediately as an input to the reactive ANN. In the parallel mode active and reactive networks process a sample of measurements independently (the result of active network is not used as an immediate input to the reactive network). Parallel operation can increase the speed of computation. However, most of the computing time is used to calculate the residuals given by Eqs. (3.8) and (3.9). To speed up the computation, the residuals can be calculated as a difference between two consecutive samples, i.e.

$$\Delta \mathbf{Z}_P = \mathbf{Z}_P(q+1) - \mathbf{Z}_P(q), \tag{3.10}$$

and

$$\Delta \mathbf{Z}_Q = \mathbf{Z}_Q(q+1) - \mathbf{Z}_Q(q). \tag{3.11}$$

The ANN SE speed of processing is controlled by the time constants τ_p and τ_q, associated with active and reactive networks, respectively. In the static state estimation only one sample of measurements is processed and it is possible to increase the speed of calculation by using small time constants (for example $10^{-6}s$). In tracking state estimation, the ANN SE uses more than one sample of measurements, so that the processing time is longer than in the static case. The time constants τ_p and τ_q are always larger than the time constant of the sample fluctuation τ_s. In fact, the result of the tracking estimator is obtained from the measurements averaged over a number of samples. The larger values for the time constants τ_p and τ_q implies a larger number of samples to be processed for one estimate. The evident advantage of the tracking state estimation is improved filtering of the Gaussian noise. The hyperbolic-tangent activation function (with properly chosen limits and slope constant) can also improve the filtering of the ANN SE in tracking mode of operation, as shown in the simulation study presented further on in this chapter.

3.2 Decoupled ANN State Estimator with Hopfield-style Dynamic

The Continuous Time Recurrent Neural Network (CTRNN) (2.57),which is one version of the Hopfield-style ANN, can be used as a power system state estimator. This was shown in Chapter 2. The weights in CTRNN do not correspond directly to the approximated Jacobian matrix (3.1), as in the steepest descent ANN SE. The Real Time Recurrent Learning (RTRL) method, presented in Chapter 2 (equations (2.61) and (2.62)) has to be used for learning the weights. Assuming weak coupling between $P-V$ and $Q-\theta$ in the power system model [10], the weight matrix is decoupled into active and reactive parts. The complete set of decoupled ANN equations (including RTRL) is:

- $P\theta$ (active) ANN with corresponding RTRL ANN

$$\tau_p \frac{d\Delta\theta_i}{dt} = -\Delta\theta_i + f_i\left(\sum_{j=1}^{N} W_{Pij}\Delta\theta_j + U_{Pi}\right), \qquad i = 1\ldots N, \tag{3.12}$$

$$\tau_y \frac{dy_{pi}}{dt} = -y_{pi} + f_i'\left(\sum_{j=1}^{N} W_{Pij}\Delta\theta_j + U_{Pi}\right)\left(\sum_{j=1}^{N} W_{Pji}y_{pj} + T_{Pi}\right), \qquad i = 1\ldots N, \tag{3.13}$$

$$\tau_w \frac{dW_{Pij}}{dt} = y_{pi}\Delta\theta_j, \qquad i,j = 1\ldots N, \tag{3.14}$$

- QV (reactive) ANN with corresponding RTRL ANN

$$\tau_q \frac{d\Delta V_i}{dt} = -\Delta V_i + f_i\left(\sum_{j=1}^{N} W_{Qij}\Delta V_j + U_{Qi}\right), \qquad i = 1\ldots N, \tag{3.15}$$

$$\tau_y \frac{dy_{qi}}{dt} = -y_{qi} + f_i'\left(\sum_{j=1}^{N} W_{Qij}\Delta V_j + U_{Qi}\right)\left(\sum_{j=1}^{N} W_{Qji}y_{qj} + T_{Qi}\right), \qquad i = 1\ldots N, \tag{3.16}$$

$$\tau_w \frac{dW_{Qij}}{dt} = y_{qi}\Delta V_j, \qquad i,j = 1\ldots N. \tag{3.17}$$

The inputs to $P\theta$ and QV ANN SE (Eqs. (3.12) and (3.15)) are

$$U_{Pi} = \sum_{j=1}^{M_P} H_{P\theta ji} \Delta z_{pj}, \tag{3.18}$$

and

$$U_{Qi} = \sum_{j=1}^{M_Q} H_{QVji} \Delta z_{qj}. \tag{3.19}$$

The matrices $\mathbf{H}_{P\theta}$ and $\mathbf{H}_{QV}$ are constant matrices, and are given in Appendix A. The inputs T_{Pi} and T_{Qi} to RTRL networks (Eqs. (3.13) and (3.16)) are (Appendix C, Eq. (C.17)):

$$T_{Pi} = \sum_{j=1}^{M_P} p_{pj} r_{pj} \frac{\partial h_{Pj}}{\partial \theta_i}, \tag{3.20}$$

and

$$T_{Qi} = \sum_{j=1}^{M_Q} p_{qj} r_{qj} \frac{\partial h_{Qj}}{\partial V_i}, \tag{3.21}$$

where p_{pj} and p_{qj} are the weights, and r_{pj} and r_{qj} are the measurement residuals. $\frac{\partial h_{Pj}}{\partial \theta_i}$ and $\frac{\partial h_{Qj}}{\partial V_i}$ are the elements of the Jacobian matrix given in Appendix A. The approximated constant values for $\frac{\partial h_{Pj}}{\partial \theta_i}$ and $\frac{\partial h_{Qj}}{\partial V_i}$ in matrices $\mathbf{H}_{P\theta}$ and $\mathbf{H}_{QV}$ also can be used in the Decoupled ANN SE. They are given in Appendix A.

Synthesis of the estimator for a specific power system and range of operations of that system is the first necessary step in using the CTRNN estimator. The RTRL algorithm, implemented on an ANN, has to be used to learn the weights of the CTRNN estimator. Learning can be performed off-line on simulated data. The second step is the application of the synthesized CTRNN in state estimation.

Similar to the steepest descent ANN SE, the CTRNN estimator given by Eqs. (3.12) and (3.15) can be used both in the static and the tracking mode of operation.

3.3 Simulation Study

3.3.1 Static State Estimation

The IEEE 14-bus power system is used for the simulation study of the Decoupled ANN SE with steepest descent dynamic used in static state estimation. The algorithm was presented in Section 3.1.1. Power system parameter data, initial values and metering locations are given in Appendix D. First, we solve the load flow problem for initial load values. The obtained results are used in order to find measurement values. The measurement set is composed of 30 active and 30 reactive power measurements, and 7 voltage magnitude measurements, as shown in Appendix D. Measurement errors are simulated as normally distributed random values with standard deviations $0.007pu$ for voltage magnitude, and $0.02pu$ for power measurements. The Runge-Kutta method is used to simulate ANN in steps 4 and 8 of the algorithm. The simulation program is written in MATLAB. The simulation CPU time on PC486, 66MHz was 256s. From the results of the simulation we plot the time evolution of the voltage phase angle and the voltage magnitude at node 2 in Figures 3.1 and 3.2 respectively.

The algorithm converged in three iteration steps. At each iteration step the objective functions of active ANN, and reactive ANN are minimized. Sequential operation of the active and the reactive networks creates sudden jumps in angle and magnitude values during computation, as shown in Figures 3.1 and 3.2. The activation function for all nodes in both networks is the same: the hyperbolic-tangent with the parameters $L_u = -L_l = 1$ and $c = 1$.

It is possible to implement the active and the reactive networks by using the neuron circuit from Figure 2.17 for each of the activation nodes. If the neuron resistance $R = 1k\Omega$ and capacitance $C = 1nF$ are used, the time constants become $\tau_p = \tau_q = 10^{-6}s$. The time constants are calculated by using equation (2.65). If

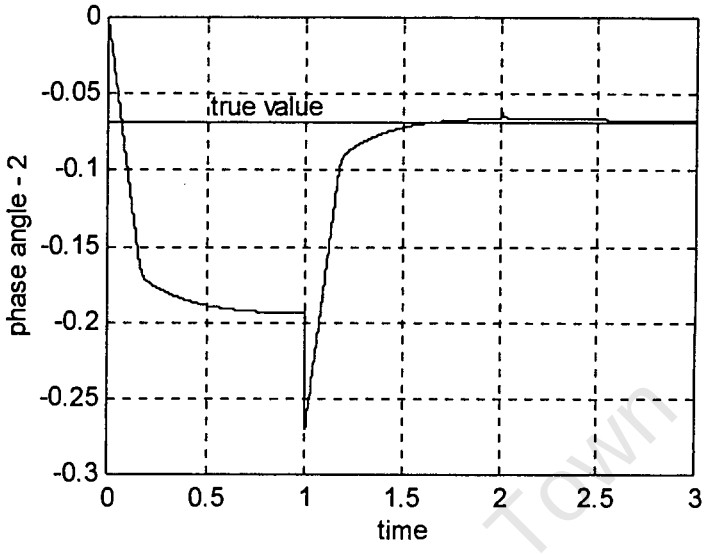

Figure 3.1: *Time evolution of the voltage phase angle at node 2*

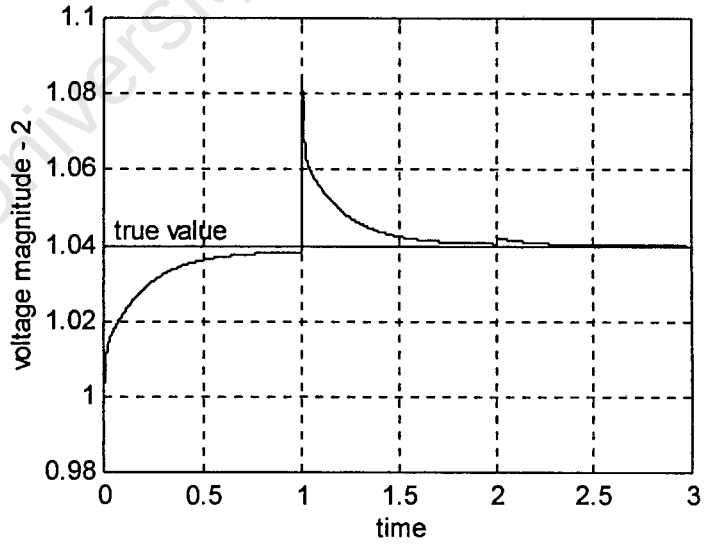

Figure 3.2: *Time evolution of the voltage magnitude at node 2*

such a hardware is used in this example, the unit for time in Figures 3.1 and 3.2 will be $10^{-6}s$. It can be predicted that the speed of computation for the steps 4 and 8 will be $3\mu s$ on analog ANN hardware. The computation speed is the same for larger problems because the neurons are operating in parallel. However, to find the total computing time we need to take into account the time needed for steps 2, 3, 6 and 7 of the algorithm (calculation of the residuals). We can estimate that the total computing time on analog ANN hardware will be in a range of few ms.

3.3.2 Tracking State Estimation

The tracking state estimation algorithm based on ANN SE (steepest descent and Hopfield-style versions), presented at the beginning of this chapter, is compared with the tracking Fast Decoupled State Estimator (FDSE) [78]. The IEEE 14-bus test system was used in the simulation study. All detail of the power system and the measurements (a total of 67 measurements) are given in Appendix D.

The simulation study is done over 35 time-sample intervals. All loads vary with the same pattern, as shown in Figure 3.3. For each load, the active power and the reactive power for the sample q are given with $P(q) = sP_I$ and $Q(q) = sQ_I$ respectively, where s is the rate of change. The initial values P_I and Q_I are given in Appendix D.

The load pattern consists of a slowly varying part (from $q = 1$ to $q = 30$) and a sudden change in the operating point (from $q = 30$ to $q = 32$). Such a pattern corresponds to conditions encountered in a power system.

Exact values of the measurements have been calculated by running successive load flows for each change of load. Measurement errors with the specification

- $\sigma = 0.007pu$ for voltage magnitude measurements, and
- $\sigma = 0.02pu$ for power measurements,

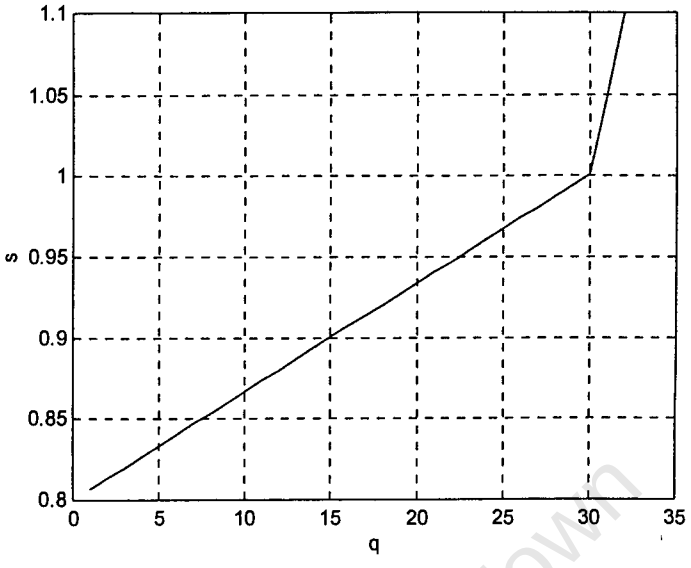

Figure 3.3: *Load pattern $s(q)$; $P(q) = s(q)P_I$ and $Q(q) = s(q)Q_I$; P_I and Q_I are initial values*

have been simulated by using a random number generator, and were then added to the measurements for each sample.

In this simulation study the filtering performance of a tracking ANN SE is assessed by using the performance index given in the following formula:

$$PI = \frac{\sum_{i=1}^{M} \left(z_i^E - z_i^T\right)^2}{\sum_{i=1}^{M} \left(z_i^M - z_i^T\right)^2}, \qquad (3.22)$$

where z_i^T, z_i^M and z_i^E are the true, measured and estimated values of the $i-th$ measurement. The numerator of the performance index indicates how close the estimated values are to the true values, and the denominator shows the level of uncertainty of the measurements. In the case of good filtering, the performance index should always be smaller than one. According to the P, Q, V and θ measurements and formula (3.22), the following indices are introduced: PI_P, PI_Q, PI_V and PI_θ. All of these indices have been calculated for each sample and then averaged over 30 samples (slow varying operating point) and over 35 samples (including the sudden change of

the operating point). The standard deviation of the error in angle estimation (σ_θ) and the standard deviation of the error in voltage magnitude estimation (σ_V) have been computed from simulated data (for two cases: 30 and 35 samples).

In this study we compare the performance of tracking FDSE and ANN based tracking state estimation (steepest descent and Hopfield-style versions) in the following modes of operation:

- **Method 1** - sequential calculation of active and reactive ANN;
- **Method 2** - parallel calculation of active and reactive networks; measurement residuals are calculated using equations (3.8) and (3.9);
- **Method 3** - parallel and completely independent calculation of active and reactive networks; measurement residuals are calculated using equations (3.10).

The continuous time ANN SE is transformed to the discrete time ANN SE, as given in the Section 2.5.2. The simulations were performed by using discrete ANN SE methods. For the gain constants of the discrete models we use $\mu_p = 1$ and $\mu_q = 1$ (time constants: $\tau_p = \tau_q = 1$). The limits of the hyperbolic tangent activation functions in all examples are chosen to be $L_u = -L_l = 0.02$. All simulation programs are written in MATLAB.

ANN SE with Steepest Descent Dynamic versus Tracking FDSE

In Tables 3.1, 3.2, and 3.3 the results of the simulation of ANN SE with steepest descent dynamic in tracking mode of operation, and the simulation of tracking FDSE are summarized. The hyperbolic tangent activation function slope constants used are $c_\theta = 50$ (for active ANN) and $c_V = 25$ (for reactive ANN). The CPU times for processing one sample of measurements on PC486, 66MHz are as follows: 2.6s - tracking FDSE, 2.9s - **Method 1**, 2.5s - **Method 2**, 1.9s - **Method 3**.

	PI_P35	PI_P30	$PI_\theta 35$	$PI_\theta 30$
Tracking FDSE	0.7477	0.7002	0.5302	0.5473
Method 1	*0.7389	0.7075	0.3706	0.3884
	**0.6537	0.6227	0.3678	0.3775
Method 2	*0.7714	0.738	0.3031	0.3055
	**0.6072	0.5759	0.3678	0.3775
Method 3	*0.9281	0.8206	0.3785	0.3778
	**0.7778	0.6913	0.3984	0.3654

Table 3.1: *Performance of the ANN SE with steepest descent dynamic compared to FDLF - Performance indices for active measurement set (* linear activation and ** hyperbolic tangent activation)*

	PI_Q35	PI_Q30	PI_V35	PI_V30
Tracking FDSE	0.6238	0.6265	0.531	0.5267
Method 1	*0.6136	0.6136	0.5537	0.543
	**0.3965	0.3899	0.251	0.2537
Method 2	*0.7172	0.724	0.6015	0.6022
	**0.481	0.4749	0.3415	0.3415
Method 3	*1.1037	1.0028	0.587	0.5551
	**0.9264	0.7977	0.4127	0.3632

Table 3.2: *Performance of the ANN SE with steepest descent dynamic compared to FDLF - Performance indices for reactive measurement set (* linear activation and ** hyperbolic tangent activation)*

	$\sigma_\theta 35$	$\sigma_\theta 30$	$\sigma_V 35$	$\sigma_V 30$
Tracking FDSE	0.0032	0.0031	0.0039	0.0037
Method 1	*0.0029	0.0029	0.0034	0.0033
	**0.0029	0.0029	0.0016	0.0016
Method 2	*0.0034	0.0031	0.0035	0.0036
	**0.0031	0.0031	0.0021	0.0021
Method 3	*0.0036	0.0035	0.0037	0.0034
	**0.0036	0.0032	0.0039	0.003

Table 3.3: *Standard deviations of state estimate errors (* linear activation and ** hyperbolic tangent activation)*

From the presented results we conclude that the ANN State Estimator (**Method 1**) with steepest descent dynamic gives better results in terms of filtering of white noise, as compared to the tracking FDSE. Filtering characteristic of the ANN State Estimator can be further improved by using larger time constants. Then the response of ANN is slower and more samples of measurements are averaged for one estimate. During the moments of sudden change of the operating point, slow response of ANN SE is not a desirable characteristic. A few different time constants can be used for the ANN SE, depending on the rate of change of load in the daily load cycle.

The simulation results show that the hyperbolic tangent activation function can improve the filtering performance of the ANN State Estimator. The upper and the lower limits control the maximum change per one sample. The slope constant can also influence the performance. In the presented simulation study the slope constants were large, so that the activation was close to the hard limiting non-linearity. Sensitivity analysis presented in the next section demonstrates the influence of the slope constant on filtering performance of an ANN SE.

The ANN State Estimator (**Method 3**) with hyperbolic tangent activation function produces acceptable filtering performance, and it is the fastest computation technique on sequential and parallel hardware. However, the result tends to degenerate in the case of a rapid change in operating point. Then **Method 1** or **Method 2** should be used.

ANN SE with Hopfield-style Dynamic versus Tracking FDSE

In Tables 3.4, 3.5, and 3.6 the results of the simulation study of ANN SE with Hopfield-style dynamic (CTRNN) in the tracking mode of operation are summarized, and compared to the tracking FDSE. In this study the hyperbolic tangent activation was used. The activation function slope constants are $c_\theta = 1$ (for active ANN) and $c_V = 1$ (for reactive ANN). Before we tested the estimator the weights have

	PI_P35	PI_P30	$PI_\theta35$	$PI_\theta30$
Tracking FDSE	0.7477	0.7002	0.5302	0.5473
Method 1	0.4274	0.415	0.2558	0.2648
Method 2	0.8957	0.7493	0.8176	0.5377
Method 3	3.0729	2.3801	2.5386	2.1569

Table 3.4: *Performance of the CTRNN State Estimator; Performance indices for active measurement set*

	PI_Q35	PI_Q30	PI_V35	PI_V30
Tracking FDSE	0.6238	0.6265	0.531	0.5267
Method 1	0.3533	0.3112	0.2573	0.2553
Method 2	0.4725	0.3967	0.2649	0.2483
Method 3	1.2956	1.0133	0.3367	0.2140

Table 3.5: *Performance of the CTRNN State Estimator; Performance indices of reactive measurement set*

been found through the learning process by using the RTRL technique. The program for RTRL is written in MATLAB. Learning has been performed on simulated data (obtained through load flow calculation) for all 35 time-samples covering the load change shown in Figure 3.3. The active and the reactive networks are learned sequentially (**Method 1**). The sum of square errors between CTRNN estimator output and specified true values is minimized to 1.58×10^{-5} (for active ANN), and 4.6×10^{-5} (for reactive ANN) over 35 operating points. The CPU time for learning on PC486, 66MHz was 1762s. The estimator is trained for these 35 points covering the range of operation, and can not be used for some other values out of that range without previous re-training (adaptation). The RTRL technique is computationally intensive and very slow on sequential hardware. However, the RTRL method is designed for parallel implementation on special-purpose hardware [51]. With such a hardware the learning is more efficient, and the method can be utilized in on-line weights adaptation.

The results presented in Tables 3.4, 3.5, and 3.6 indicate that CTRNN estimator (**Method 1**) has a better filtering performance compared to the tracking FDSE.

	$\sigma_\theta 35$	$\sigma_\theta 30$	$\sigma_V 35$	$\sigma_V 30$
Tracking FDSE	0.0032	0.0031	0.0039	0.0037
Method 1	0.003	0.003	0.0017	0.0016
Method 2	0.0043	0.0025	0.0021	0.0017
Method 3	0.0204	0.014	0.0058	0.0042

Table 3.6: *CTRNN State Estimator: Standard deviations of the estimation errors*

The simulation times are the same as times mentioned in the previous section for steepest descent ANN SE and tracking FDSE.

In this study the learning has been done assuming **Method 1**. This is the reason why **Method 2** and **Method 3** are not giving good results (some of the performance indices are larger than one). However, it is possible to train active and reactive networks using these two methods, and then reach acceptable filtering performance.

ANN SE with Steepest Descent Dynamic versus ANN SE with Hopfield-style dynamic

In this section, tracking estimators with steepest descent dynamic and Hopfield-style dynamic (CTRNN) in sequential mode of operation (**Method 1**) are compared by using the performance index (3.22) and sensitivity measure. For the calculation of the sensitivity measure in the steepest descent type of ANN estimator we use formula (B.43); for Hopfield-style ANN estimator we use formula (B.49). The Hopfield-style ANN SE with slope constants $c_\theta = c_V = 1$ was learned before. Now we again learn the same ANN SE but for slope constants $c_\theta = c_V = 0.5$ and for 35 different conditions given by the load pattern represented in Figure 3.3. In the learning phase we use measurements that are not corrupted by errors. In the learning process the sum of square errors for active and reactive ANN estimators is minimized to 10^{-5}. In the learning phase and simulations we used the gain constants $\mu_p = \mu_q = 1$, and activation function limits $L_u = -L_l = 0.02$. The results (performance indices:

	active measurements			reactive measurements		
steepest descent dynamic		$\bar{S}_P$	PI_P30		$\bar{S}_Q$	PI_Q30
	$c_\theta = 1$ $c_V = 1$	0.2236	0.1299	$c_\theta = 1$ $c_V = 1$	0.1581	0.0912
	$c_\theta = 0.5$ $c_V = 0.5$	0.0707	0.0186	$c_\theta = 0.5$ $c_V = 0.5$	0.05	0.0118
Hopfield-style dynamic (CTRNN)		$\bar{S}_P$	PI_P30		$\bar{S}_Q$	PI_Q30
	$c_\theta = 1$ $c_V = 1$	0.2772	0.415	$c_\theta = 1$ $c_V = 1$	0.2531	0.3112
	$c_\theta = 0.5$ $c_V = 0.5$	0.1942	0.2606	$c_\theta = 0.5$ $c_V = 0.5$	0.1774	0.1962

Table 3.7: *Sensitivity measure and performance index for ANN SE*

PI_P30 and PI_Q30; and sensitivity measures: $\bar{S}_P$ and $\bar{S}_Q$) are summarized in Table 3.7. The simulation CPU times are mentioned in the previous two sections.

For this case the results indicate that the ANN SE with steepest descent dynamic has a better filtering performance compared to the ANN SE with Hopfield-style dynamic. We conclude the following:

- The sensitivity measure corresponds to the performance index of the ANN SE (steepest descent and Hopfield-style versions) in tracking mode of operation when the measurement errors are small and modelled as a white noise sequence.
- The ANN SE (steepest descent and Hopfield-style versions) in tracking mode of operation with smaller slope constant is less sensitive to white noise errors in the measurements.
- It is possible to use the sensitivity measures in an ANN SE (steepest descent and Hopfield-style versions) design to predict the performance for different sets of weights and different parameters of the activation function.

Chapter 4

Phasor Angle Measurements

It is presumed that the voltage phasor angle measurements used together with voltage magnitude, active and reactive power flow and injection measurements would improve the performance of Decoupled ANN SE (Eqs. (3.2) and (3.3)). For the voltage phasor angle measurements a special instrument, the so called Phasor Measurement Unit (PMU), is used [82]. In the first section of this chapter an overview of the PMU instrument is presented. There are a few alternatives in which the phasor angle measurements could be used in a Decoupled ANN SE. These alternatives are discussed in the following subsections. At the end of the chapter the results of the sensitivity analysis and the simulation study are presented. These results clearly show the benefits of using phasor angle measurements in the ANN SE.

4.1 Phasor Measurement Unit

4.1.1 Design and Operation

Design of the PMU is completely different to the classical measuring element [70]. The main functional blocks of a PMU are [83]:

- Input signal conditioning circuit (interfaced with a CT and a VT).
- Analog to digital (A/D) converters.
- Master clock generator and Global Positioning System (GPS) receiver.

- Host processor.
- Communication interface.

In the classical measuring systems transducer designs are based on analog techniques [70]. The PMU technology uses a different approach; the signal conditioning circuit interfaces directly to the VT and CT. It performs current to voltage conversion, and creates voltage signals (using insulation transformers) suitable for conversion to the digital form. It also does the required filtering at the Nyquist rate. Then the phasor values (magnitude and angle) of positive sequence voltage and current, as well as MW and MVAr, are calculated in software run by the host processor of the instrument.

The most recent design makes use of a high quality active instrumentation amplifier followed directly by the A/D converter [83]. Isolation is achieved by optical coupling of the A/D outputs and control signals. Power is supplied to the "live" circuit by means of an isolated switch mode power supply. This design achieves a minimum, common mode rejection ratio of roughly 90 dB across the operating spectrum, and an A/D converter dynamic range in access of 80 dB. This excellent signal conditioning quality is very important for the applications when voltage and current signal measurements during the dynamic response of a power system are required. The ANN SE using parallel hardware, in analog (Section 2.5.1) or digital (Section 2.5.2) form, is capable of tracking the dynamic response of a power system in real-time. However, this quality of signal conditioning is not necessary when the ANN SE operates in the static mode, calculating state estimates every few minutes from available snapshot of the measurements.

After conditioning, voltage and current input signals are sampled (usually 12 or 20 samples per cycle of the fundamental frequency) by using a GPS co-ordinated clock. Each sample will receive a time-tag from the GPS receiver. These tags are

used to synchronize the measurements received from the PMU's across the power network.

PMU technology provides automatic transformation of the sampled time-domain measurements into a frequency domain. The transformation takes the samples of voltages and currents from all three phases into the positive sequence values. One transformation used in the existing instrument [83] is the recursive Discrete Fourier Transform (DFT) [82]. There are other transformations available. The comparison study of these transformations is available in reference [84]. The recursive DFT algorithm calculates positive sequence, fundamental frequency voltage and current phasors for each sample of measurements. By using these phasors the local fundamental frequency, and the rate of change of the fundamental frequency can be calculated. The rate of change of magnitude and phasor angle is also available.

Recursive DFT is developed from the full cycle DFT. It means that the transformation will filter out harmonics (and DC offsets) and give as an output only a fundamental frequency. It is clear that the algorithm will need a full cycle to recover after a switching disturbance. A sudden change of phasor angle is possible during switching transients. The algorithm needs to be initialized again after switching transients. This should be done simultaneously for all PMU's in the network.

A large error in the result of the recursive DFT is possible during large excursions of local fundamental frequency ($\pm 5Hz$). In this situation results need to be corrected which requires an additional computation and a resultant slower response. For more common excursions of $\pm 1Hz$ the errors can be safely ignored.

4.1.2 Measurement Errors

In general, the accuracy of the power system measurements depends on:

- the accuracy of the voltage and current transformers and transducers used,

- the accuracy of the analog to digital equipment used,
- the level of harmonics present,
- the level of imbalance, resulting from untransposed transmission lines and/or unbalanced loads.

The errors in current and voltage instrument transformers are present in both the classical measuring elements and in the PMU technology. These errors can be minimized through calibration [70].

Any harmonic interference present in the power system will affect the accuracy of the RMS voltage, and active and reactive power measurements obtained from the classical analog transducers based on rectification of AC signals. This is not the case with the PMU technology where recursive DFT filters out harmonics.

The single phase (positive sequence) network model, used by an ANN SE, assumes that no imbalance exists and that the three phase system can be represented by a single phase equivalent. In practice, power systems are unbalanced, when the transmission lines are untransposed and the loads not balanced. It is common that a large number of voltage measurements are detected as being faulty, while only a few faulty active and reactive power measurements are detected. This problem is due to the power system imbalance. Usually only one voltage (from one phase) is sent to the control centre, whereas the active and reactive power measurements are obtained from different combinations of voltages and currents from two or three phases. The PMU technology helps in solving the problem with unbalanced system state estimation by using sequence component transformation on all three phase voltages and currents. The positive, negative and zero sequence measurements are the outputs from PMU to be used in state estimation algorithm which must use power system model for all three sequences.

The error model used for classical RMS measurements assumes an error that depends upon both the magnitude of the measurement and the full-scale value of the instrument. The phasor measurements error model depends on full-scale rating and A/D resolution. For example, if the full-scale input is 1.5 times the nominal input and a 1% maximal error is required, the minimum A/D resolution should be 9 bits. For line current measurements the full-scale rating can be large (3 times the nominal). In that case, for a 1% accuracy, the A/D resolution should be 13 bits. For phasor angle measurements a tenth of a degree, at the power system frequency provides sufficient accuracy. This can be achieved with the $1\mu s$ accuracy available from GPS receivers. Errors in system synchronization, sampling, and data window all must be included. In order to complete the error model CT and VT errors, and anti-aliasing errors need to be added. All these small errors are modeled as normally distributed random values with a known standard deviation.

4.2 Use of Absolute Voltage Phasor Angle Measurements

If some bus voltage angles θ_i are measured, the active measurement vector $\mathbf{Z}_P$ becomes

$$\mathbf{Z}_P = \begin{pmatrix} Pf_{ij}/V_i & P_i/V_i & \theta_i \end{pmatrix}^T. \qquad (4.1)$$

The expanded measurement vector (4.1) is used as an input to the active part of the Decoupled ANN SE (Eq. (3.2)). The linear function relating the measured angles to the state variables is simply

$$\theta_i = \theta_i.$$

Modification of the Decoupled ANN SE dynamical equations (3.2) is straightforward. The entries of the Jacobian sub-matrix $\mathbf{H}_{P\theta}$ related to angle measurements are equal to one. A new $\mathbf{H}_{P\theta}$ implies a change in the weight matrix $\mathbf{W}_P$ and input vector $\mathbf{U}_P$ (Eqs. (3.4) and (3.5)).

A very important issue is referencing the voltage phasor angle measurements. One of the power system buses is chosen as the reference in the active ANN SE (Eq. (3.2)). All estimates of the voltage phasor angles at the system buses are with respect to this chosen reference. Direct angle measurements all have different references, determined by the instant the sampling is initiated on PMU instruments. If the sampling is synchronized on all instruments, we have only one reference but this reference is different from the estimator reference. If direct absolute angle measurements are used without dealing with the reference problem, unrealistic results are to be expected.

A possible solution is to install a PMU instrument at the reference bus of the ANN SE. The angle measured at the reference should then be subtracted from all the other synchronized direct angle measurements. In this solution, the failure of the instrument on a reference bus can be a problem. All the direct absolute angle measurements become useless. However, this failure is easy to detect. For example, detection could be based on the large difference between existing estimated angle values for the previous sample of measurements and all the angle measurements of the actual sample. In this case a simple change of reference in the ANN SE will solve the problem. The new reference bus should of course be equipped with a PMU instrument.

Another possible way of dealing with the different references is to use the ANN SE and not to use, as an input, the direct absolute angle measurements. In this case the ANN SE produces an estimate of the voltage angles for the buses where the PMU instruments are installed. Direct measurement of the same angles with the time synchronizing reference is available from PMU instruments. The average difference between the measured and estimated angles is an adequate estimate of the difference between the two references.

Testing of the classical WLS Static State Estimator (Fast Decoupled algorithm) including the phasor angle measurements, on the IEEE 118 bus test system, is reported in reference [85]. The standard deviation for simulated Gaussian errors added to the angle measurements vary between 0.1° and 5°. A number of cases with different metering configurations and measurement accuracies are studied by means of the quasi Monte Carlo simulation. The following conclusions gained from this study are valid also for the Decoupled ANN SE in the static mode of operation:

- If the angle measurements are sufficiently accurate (standard deviation less than 2°) improvement in the convergence speed is likely to occur (the number of iterations is reduced); even with larger errors in angle measurements (standard deviation is 5°) there are no convergence problems.

- The angle measurements must be very accurate (standard deviation close to 0.1°) if it is not to have an adverse effect on noise filtering when combined with the measurements of power flows, injections, and voltage magnitudes.

- If the power flow measurements are replaced with voltage phasor angle measurements the filtering performance of the estimator is degraded.

- The estimator based on a large number of voltage magnitude and angle measurements and just a few power flow and injection measurements will have only limited ability to deal with bad data in the conventional manner. On the other hand, because of low local redundancy, a bad measurement has no effect on the estimate of the remaining states of the network.

4.3 Use of Angle Difference Between Voltage Phasor Measurements

A voltage phasor angle measurement has an arbitrary chosen reference. Such a measurement has no physical meaning. As shown before, these measurements can be used directly in the ANN SE, but, the problem of different references between ANN estimator and angle measurements must be solved. A different approach, where the problems with the reference do not exist, is to use the angle difference between voltage phasors at both ends of transmission line or transformer. Simply stated, two voltage phasor angle measurements (synchronized) obtained from two PMU instruments are subtracted from each other.

In this case the active measurement vector $\mathbf{Z}_P$ becomes

$$\mathbf{Z}_P = \begin{pmatrix} Pf_{ij}/V_i & P_i/V_i & \theta_{ij} \end{pmatrix}^T. \tag{4.2}$$

The function relating measured phasor shift θ_{ij} with state variables is

$$\theta_{ij} = \theta_i - \theta_j$$

The modification of the active part of Decoupled ANN SE (Eq. (3.2)) is very simple. There are two new entries (equal to 1 and -1) in the Jacobian sub-matrix $\mathbf{H}_{P\theta}$ related to one angle shift measurement (Appendix A). Change in $\mathbf{H}_{P\theta}$ implies that the weight matrix $\mathbf{W}_P$ and the input vector $\mathbf{U}_P$ (Eqs. (3.4) and (3.4)) should be changed.

In this approach the problem with reference does not exist. The phasor angle shift does not depend on reference. It is only important that the measurement samples of the bus voltage phasors are accurately synchronized.

4.4 Simulation Study

The performance of Decoupled ANN SE with steepest descent dynamic (Eqs. (3.2) and (3.3)) in static and tracking modes of operation (Paragraphs (3.1.1) and

(3.1.2)) when the voltage phasor angle shift measurements are included, is evaluated through simulation and sensitivity studies. The voltage phasor angle shift measurement has an advantage over absolute phasor angle measurement because the reference problem is avoided.

The 14-bus test system has been used in this study. All the detail on the power system and the set of measurements are given in Appendix D. The angle shift measurements are added to the classical set of measurements. Four cases are studied in the following sections:

- **Case 1**: 3 angle measurements; lines 1-2, 4-9, 5-6;
- **Case 2**: 7 angle measurements; lines 1-2, 2-4, 2-3, 4-9, 5-6, 6-13, 10-11;
- **Case 3**: 12 angle measurements; lines 1-2, 1-5, 2-3, 2-4, 2-5, 3-4, 4-5, 4-7, 4-9, 5-6, 6-11, 6-12;
- **Case 4**: 19 angle measurements; all lines.

Measurement errors are simulated by using a random number generator with normal distribution, according to the specification: $\sigma_V = 0.007pu$ (voltage magnitude measurements) and $\sigma_p = \sigma_q = 0.02pu$ (active and reactive power measurements). Two values are used for standard deviation of the angle shift measurements: $\sigma_\theta = 0.00174rad$ (0.1°) and $0.0873rad$ (5°). These errors are added to measurement values obtained from the load flow study.

4.4.1 Static State Estimation

Decoupled ANN SE with steepest descent dynamic is used as a static state estimator in the manner explained in Paragraph 3.1.1. The parameters of the hyperbolic-tangent activation function are: $L_u = -L_l = 1$ (upper and lower limits) and $c = 1$ (slope constant). The time constants τ_p and τ_q depend on how fast is

the hardware, on which the ANN SE is implemented. In the static state estimation the time constants do not influence the filtering performance; they only influence the estimator response time. The impact of the angle shift measurements on ANN SE filtering of small measurement errors is studied by calculating certain sensitivity coefficients and by simulating the ANN SE for a number of different cases. The results and discussions are presented in the following two subsections.

Sensitivity study

The state estimation sensitivity matrix (Eq. (B.30), derived in Appendix B) for the **Case 1** (only active measurement set) and standard deviation of the angle shift measurements equal to 0.1° is calculated. This sensitivity matrix $\mathbf{P}_P$ relates the state estimation error and measurement error for the Decoupled ANN SE in the static mode or operation (Appendix B, Eq. B.28):

$$\boldsymbol{\Theta}^* - \hat{\boldsymbol{\Theta}} = -\mathbf{P}_P \mathbf{v}_P,$$

where $\boldsymbol{\Theta}^*$ is the vector of true states and $\hat{\boldsymbol{\Theta}}$ the vector of estimated states. One part of the sensitivity matrix is presented in Table 4.1. As one would expect, the angle estimates are most sensitive to the errors in two phasor shift measurements. The angle estimates are less sensitive to power injection and flow measurement errors.

If the state estimates are to be used for monitoring power flow on certain lines, one is concerned with sensitivities of the line flow estimates to the measurement errors. These can be computed by using the sensitivity matrix (Eq. (B.41) given in Appendix B). This sensitivity matrix $\boldsymbol{\Gamma}_P$ relates the weighted error between estimated and true measurement with the weighted measurement error (Eq. (B.40)):

$$\mathbf{R}^{-\frac{1}{2}} \left(\mathbf{h}_P \left(\mathbf{X}^* \right) - \hat{\mathbf{Z}}_P \right) = -\boldsymbol{\Gamma}_P \mathbf{v}_{WP},$$

where $\mathbf{h}_P \left(\mathbf{X}^* \right) = \mathbf{Z}_{true}$ is the vector comprising the active set of true measuring values including flows, and $\hat{\mathbf{Z}}_P$ is the vector of estimated values. Table 4.2 presents

bus\measurements	P_1	P_3	P_{1-2}	P_{2-3}	θ_{4-9}	θ_{5-6}
$\partial\theta_2$	0.0253	0.0006	0.0239	-0.0027	-0.0052	0.0004
$\partial\theta_3$	0.0304	-0.0621	0.0148	0.037	0.0013	0.004
$\partial\theta_4$	0.0339	-0.008	0.0086	0.0172	0.0561	0.0124
$\partial\theta_5$	0.0358	-0.0035	0.0052	0.0161	0.0308	0.0026
$\partial\theta_6$	0.0356	-0.0026	0.0054	0.0166	0.1428	0.4233
$\partial\theta_7$	0.0355	0.0027	0.0056	0.0187	0.3626	0.089
$\partial\theta_8$	0.0355	0.0027	0.0056	0.0187	0.3626	0.089
$\partial\theta_9$	0.0353	0.0003	0.006	0.0182	0.4082	0.1244
$\partial\theta_{10}$	0.0353	0	0.006	0.0181	0.4467	0.1437
$\partial\theta_{11}$	0.0355	-0.0013	0.0057	0.0173	0.2983	0.2802
$\partial\theta_{12}$	0.0356	-0.0024	0.0054	0.0167	0.1739	0.3979
$\partial\theta_{13}$	0.0356	-0.0022	0.0055	0.0168	0.1929	0.3824
$\partial\theta_{14}$	0.0354	-0.0008	0.0058	0.0176	0.3543	0.2259

Table 4.1: *State estimation sensitivity to measurement errors for the 14-bus system*

flow\measurement	P_{1-2}	P_{2-3}	θ_{4-9}	θ_{5-6}
∂P_{5-1}	0.0232	0.0721	0.0235	-0.002
∂P_{5-2}	-0.1078	0.1082	0.0352	-0.003
∂P_{9-4}	-0.0046	0.0017	0.1076	0.0342
∂P_{3-4}	0.0361	0.1158	-0.0545	-0.0084
∂P_{5-6}	-0.0009	-0.002	-0.0755	-0.2873
∂P_{6-11}	-0.0016	-0.0038	-0.1329	0.1222

Table 4.2: *Active power flow sensitivity to measurement errors for the 14-bus network*

one part of the sensitivity matrix Γ_P. These sensitivity coefficients show that the estimates of the line power flows are equally sensitive to power and àngle measurement errors.

Simulation results

The observations obtained by studying the sensitivity coefficients of the static ANN SE are supplemented in this section by the simulation study results. The simulation results were obtained by using 100 runs of the quasi Monte Carlo method. In this study, in order to assess the influence of the angle shift measurements on

the estimation quality of the ANN SE, we used three performance parameters as explained in this subsection.

The first performance parameter PIP is used to assess the influence of the angle shift measurements on the active power flow estimation. This performance parameter is given in the following formula

$$PIP = \frac{\sum_{i=1}^{m_{pf}} \left(Pf_i^E - Pf_i^T\right)^2}{\sum_{i=1}^{m_{pf}} \left(Pf_i^M - Pf_i^T\right)^2}, \tag{4.3}$$

where superscripts T, M, and E are the true, measured and estimated values, and m_{pf} is the number of active power flow measurements. The numerator of the performance index indicates how close the estimated values are to the true values, and the denominator shows the level of uncertainty of the measurements. In the case of good filtering, the performance index should always be smaller than 1. The PIP index has been calculated for each sample of the Monte Carlo simulation. The final value for the PIP index is the value averaged over 100 samples.

The estimated standard deviation $\hat{\sigma}_\theta$ of the error in the bus voltage phasor angle estimation, averaged over all estimates is the second performance parameter. This parameter is obtained from the simulation results, and it is the same as the one calculated by using the estimation error covariance matrix (Eq. (B.32) given in Appendix B). The third parameter is the estimated standard deviation $\hat{\sigma}_{WP}$ of the weighted measurement estimation errors, averaged over all measurements from the active set. It is obtained from the simulation results. The influence of angle shift measurements on reactive power and voltage magnitude is negligible, and does not need a special study.

The study results are summarized in four tables. Tables 4.3 and 4.4 show the trend of the performance index PIP (Eq. (4.3)), $\hat{\sigma}_\theta$, and $\hat{\sigma}_{WP}$ when we increase the number of angle measurements (covering all cases from 1 to 4). Two accuracy classes of angle measurements, 0.1° and 5° where simulated. For the same two accuracy

$\sigma_\theta = 0.1°$				
No. of θ_{ij} meas.	3	7	12	19
PIP	0.6	0.5941	0.5682	0.5351
$\hat{\sigma}_\theta$	0.08°	0.074°	0.0687°	0.063°
$\hat{\sigma}_{WP}$	0.596	0.5622	0.5211	0.4881

Table 4.3: *ANN SE performance parameters; influence of the high quality phasor angle measurements*

$\sigma_\theta = 5°$				
No. of θ_{ij} meas.	3	7	12	19
PIP	0.6516	0.6366	0.6514	0.653
$\hat{\sigma}_\theta$	0.092°	0.092°	0.092°	0.092°
$\hat{\sigma}_{WP}$	0.5755	0.5145	0.4545	0.3917

Table 4.4: *ANN SE performance parameters; influence of the low quality phasor angle measurements*

classes we simulated the effect of removing active power measurements (while keeping the maximal number of the phasor shift measurements) to estimation quality. The results are shown in Tables 4.5 and 4.6.

Conclusions

From the presented results one can conclude that the angle shift measurements can improve filtering performance of the ANN SE with steepest descent dynamic. However, these measurements can degrade the performance if they are not accurate enough. For the ANN SE active flow and injection measurements with enough redundancy are the most important for good state estimates. The angle measure-

$\sigma_\theta = 0.1°$, all phasor shift measurements included, total No. of power meas. is 30						
No of power meas.	25	21	17	10	6	0
$\hat{\sigma}_\theta$	0.063°	0.0687°	0.074°	0.085°	0.085°	0.1146°
$\hat{\sigma}_{WP}$	0.5145	0.5389	0.5653	0.6383	0.6969	0.8224

Table 4.5: *ANN SE performance parameters; influence of the active power measurements reduction while the high quality phasor angle measurements are used*

$\sigma_\theta = 5°$, all phasor shift measurements included, total No. of power meas. is 30						
No of power meas.	25	21	17	10	6	0
$\hat{\sigma}_\theta$	0.3552°	0.3781°	1.3178°	2.807°	4.034°	5.97°
$\hat{\sigma}_{WP}$	0.4148	0.421	0.4744	0.5621	0.6647	0.8224

Table 4.6: *Static ANN SE performance parameters; influence of the active power measurements reduction while the low quality phase angle measurements are used*

ments should only complement (not replaced) the active flow and the injection set of measurements. This conclusion follows from Tables 4.5 and 4.6 where the estimator performance obviously deteriorates if we take out power flow and injection measurements. If the number of angle measurements is large and the number of active power measurements low, the angle measurements need to be very accurate to obtain acceptable estimates. It can be seen from the simulation results that PIP and $\hat{\sigma}_{WP}$ are not as sensitive as $\hat{\sigma}_\theta$ is to small measurement errors, yielding the conclusion that active power flow and injection estimates are not very much affected by low accuracy of the angle measurements ($\sigma_\theta = 5°$). This was also the conclusion from the sensitivity study.

4.4.2 Tracking State Estimation

This simulation study was made in order to assess the influence of the additional set of measurements (angle shift measurements) on the filtering performance of the Decoupled ANN SE with steepest descent dynamic in tracking mode of operation, described in Section 3.1.2. The load-flow simulation of the 14-bus power system was made over 35 time-sample intervals, where all loads vary with the same pattern as shown in Figure 3.3 (in Section 3.3.2). For each sample of measurements, the state estimate is obtained by sequential processing of the active (Eq. (3.2)) and the reactive (Eq. (3.3)) ANN estimators, the same as **Method 1** presented in Section 3.3.2. The angle shift measurements are added to the classical set of measurements. Two cases are studied: **Case 1** and **Case 4**.

PI_P35	PI_P30	$PI_\theta 35$	$PI_\theta 30$
0.6573	0.6227	0.3678	0.3775
PI_Q35	PI_Q30	PI_V35	PI_V30
0.4065	0.3999	0.291	0.2937

Table 4.7: *Performance of the ANN SE with steepest descent dynamic including angle measurements;* **Case 1***: 3 angle measurements*

PI_P35	PI_P30	$PI_\theta 35$	$PI_\theta 30$
0.6447	0.5921	0.3058	0.3068
PI_Q35	PI_Q30	PI_V35	PI_V30
0.4019	0.3928	0.298	0.2928

Table 4.8: *Performance of the ANN SE with steepest descent dynamic including angle measurements;* **Case 4***: 19 angle measurements*

The parameters of the ANN SE with steepest descent dynamic stay the same as the parameters used in the study presented in section 3.3.2: $\tau_p = \tau_q = 1$ (time constants), $L_u = -L_l = 0.02$ (activation function limits), $c_\theta = 50$ (slope constant for active ANN SE), and $c_V = 25$ (slope constant for reactive ANN SE). The simulation results are summarized in Table 4.7 for **Case 1**, and in Table 4.8 for **Case 4**. To assess the performance of the ANN SE for these two cases we used the same performance indices as in Section 3.3.2.

An increase in the number of the phasor angle shift measurements used in tracking ANN SE with steepest descent dynamic can improve the noise filtering. This is the conclusion after comparison of the results of the simulation study for the case where angle measurements are not used (Tables 3.1, 3.2, and 3.3 in Section 3.3.2) with the cases where angle measurements are used (**Case 1**: Table 4.7 and **Case 4**: Table 4.8).

Chapter 5

Bad Data Processing

If the gross measurement errors and/or model errors are present in the observation equation (2.1), the ANN State Estimator will give erroneous results. In this chapter we use the relationship between the errors and the measurement residual vector in order to adapt the weights of the ANN estimator and to automatically correct the errors in the estimated state. The Real Time Recurrent Learning (RTRL) method introduced in Section 2.3.6 will be used for adaptation of weights in the ANN SE. In this case the RTRL method minimizes the sum of absolute values of residuals (discrete L_1 norm) instead of square of residuals (discrete L_2 norm).

5.1 Real Time Recurrent Learning Based on LAV Objective Function

The basic difference between Least Absolute Values (LAV) and Least Squares (LS) estimation techniques is that a LAV estimation is obtained by minimizing the sum of the absolute values of the measurement residuals whereas the LS estimation is obtained by minimizing the sum of the squares of the measurement residuals. LAV estimation is obtained by interpolating at least n out of M measurements exactly (n = the number of state variables). This is in contrast to LS estimation based on interpolation which does not necessarily pass exactly through any of the measurements.

The ANN learning technique (RTRL described in Section 2.3.6) is developed originally as a LS minimization. The objective function is (2.45)

$$E = \frac{1}{2}\sum_{i=1}^{M} p_i r_i^2, \tag{5.1}$$

where p_i represents the weighting parameter and $r_i = z_i - h_i\left(\hat{\mathbf{X}}\right)$. The vector of estimates $\hat{\mathbf{X}}$ is the output of the ANN SE. The learning algorithm adapts the weights $\mathbf{W}$ of an ANN SE by using a gradient search minimizing the objective function (5.1). Although LS provides optimal learning in the presence of noise with normal distribution and known variance, as shown in Section 2.3.6, it is unable to provide an acceptable result if any bad data is present in the measurements or in the observation model of the power system. If LAV objective function is used in RTRL, bad data will automatically be rejected in the majority of cases.

In order to adjust the RTRL to use LAV minimization criteria instead of LS, special weights are applied in Eq. (5.1). The weights change according to

$$p_i = \frac{1}{|r_i|}. \tag{5.2}$$

On convergence the objective function (5.1) becomes equal

$$E = \frac{1}{2}\sum_{i=1}^{M} \frac{1}{|r_i|} r_i^2 = \frac{1}{2}\sum_{i=1}^{M} |r_i|, \tag{5.3}$$

half of the sum of absolute values of the residuals. To avoid the convergence problems caused by division with very small numbers (residuals), a threshold r_{tol} is used, and (5.2) is modified

$$p_i = \begin{cases} \frac{1}{|r_i|} & \text{if } |r_i| \geq r_{tol}, \\ \frac{1}{r_{tol}} & \text{if } |r_i| < r_{tol}. \end{cases} \tag{5.4}$$

The threshold may be defined as a fraction (say 0.0001) of the largest residual detected in order to keep the modified weights p_i within a reasonable range.

5.2 Simulation Study

A set of simulations, using the 3-bus power system as a test system, has been devised to demonstrate the bad data processing technique for the ANN SE. This technique is based on RTRL with a LAV objective function. All details about the test power system and the associated measuring system are given in Appendix D. There are 4 real power measurements and one virtual measurement (zero injection). We assume that all measurement errors have the same standard deviation $0.01pu$.

For the 3-bus test system the dc model is considered. The model is linear and we can directly estimate the state $\mathbf{X}$ instead of the state increment $\mathbf{\Delta X}$. There are two states in the 3-bus test system: the phase angles at nodes 1 and 2. The phase angle at node 3 is the reference.

5.2.1 ANN State Estimator with Steepest Descent Dynamic

In this simulation study we have used ANN SE with steepest descent dynamic in the discrete form given by Eq. (2.68). The estimator is analytically designed by using observation matrix $\mathbf{H}$, as described in Section 2.3. The discrete form of the RTRL technique (equations 2.69 and 2.70) is used for bad data processing. As an input to RTRL we use

$$T_i = \sum_{r=1}^{M} p_r H_{ri} r_r, \qquad i = 1 \ldots n, \tag{5.5}$$

where p_r is defined in Eq. (5.4).

Gross Measurement Errors

A gross bad value for the measurement 1 is presented to the ANN SE (Eq. (2.68)): $Z_1 = -0.375$ instead of 0.375. The weights of the ANN SE (Eq. (2.68)) change automatically eliminating the influence of the bad measurement. The RTRL

technique is used for weights adaptation. The parameters of the ANN SE are: $\tau_x = 1/0.1$, $L_u = -L_l = 1$, and the slope constant $c = 1$. The parameters of the RTRL are $\tau_y = 1/0.05$, and $\tau_w = 1/0.01$. The weights p used in input T (Eq. (5.5)) change according to Eq. (5.4), so that the RTRL minimizes the sum of absolute values of measurement residuals. The influence of the bad measurement is gradually minimized and the error function is minimized accordingly, as shown in Figure 5.1. The error function is calculated using

$$J_e(t) = \left[\mathbf{X}^* - \hat{\mathbf{X}}(t)\right]^T \left[\mathbf{X}^* - \hat{\mathbf{X}}(t)\right], \tag{5.6}$$

where $\mathbf{X}^*$ is the vector of true state variables, and $\hat{\mathbf{X}}(t)$ is the state estimate vector for a specific moment in time. The time evolution of the 3-bus power system state is shown in Figure 5.2, and the time evolution of ANN SE weights in Figure 5.3.

Parameter Errors

In this example the error is simulated in the parameter B_{12} (susceptance of line $1-2$); $B_{12} = 0.1pu$ instead of $1pu$. It follows that the observation matrix changes to

$$\mathbf{H} = \begin{pmatrix} 0.1 & -1.5 & 0 & -1.5 & 1.6 \\ -0.1 & 0 & -1 & -1 & -0.1 \end{pmatrix}^T .$$

The original observation matrix is given in Appendix D. The estimation error given by Eq. (5.6) created by the wrong value of the parameter B_{12} is eliminated by using the RTRL technique, as shown in Figure 5.4. The parameters of the ANN SE (Eq. (2.68)) with RTRL (Eqs. (2.69) and (2.70)) are: $\tau_x = 1/0.1$, $L_u = -L_l = 1$, $c = 1$, $\tau_y = 1/0.01$, and $\tau_w = 1/0.001$. Time evolutions of the state and of the weights are shown in figures 5.5 and 5.6.

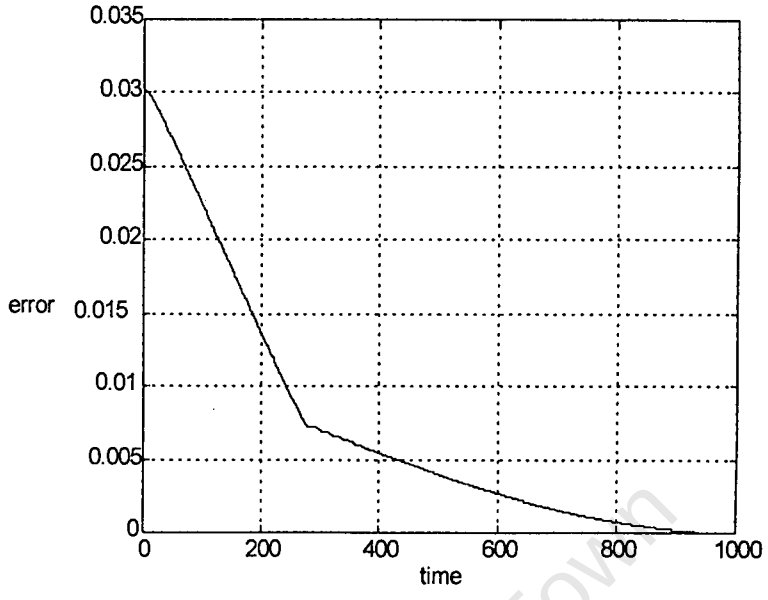

Figure 5.1: *ANN SE with steepest descent dynamic: Estimation error J_e as a function of time for the case of gross measurement error*

Topology Errors

This example shows how ANN SE (Eq. (2.68)) with RTRL (Eqs. (2.69) and (2.70)) automatically corrects estimation errors that occur when the information about line status is incorrect. In the 3-bus system information that line $2-3$ is not in operation produces the incorrect observation matrix

$$\mathbf{H} = \begin{pmatrix} 1 & -1.5 & 0 & -1.5 & 2.5 \\ -1 & 0 & 0 & 0 & -1 \end{pmatrix}^T .$$

The original observation matrix is given in Appendix D. The parameters of the ANN SE and RTRL stay the same as in the previous example. Figure 5.7 shows that RTRL gradually eliminates the estimation error given by Eq. (5.6) created by the error in topology. Time evolution of the states is shown in Figure 5.8, and time evolution of the weights in Figure 5.9.

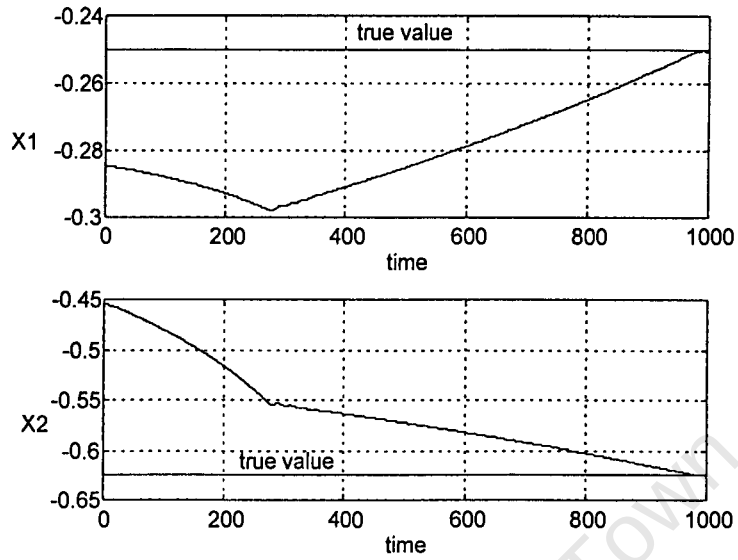

Figure 5.2: *ANN SE with steepest descent dynamic: State estimation as a function of time for the case of gross measurement error*

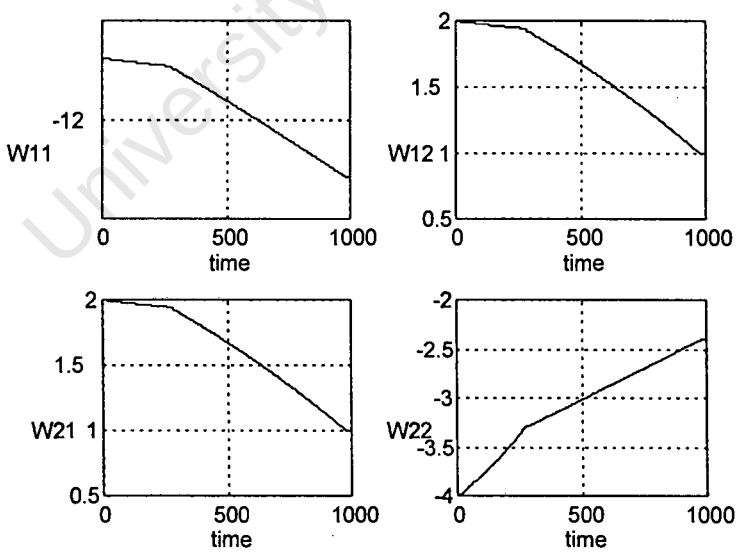

Figure 5.3: *ANN SE with steepest descent dynamic: Weights adaptation as a function of time for the case of gross measurement error*

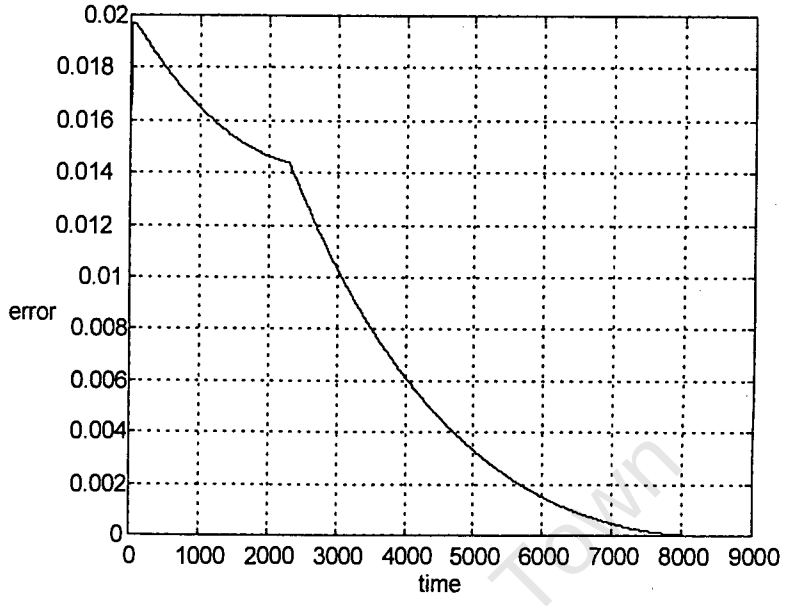

Figure 5.4: *ANN SE with steepest descent dynamic: Estimation error J_e as a function of time in the case of the parameter error*

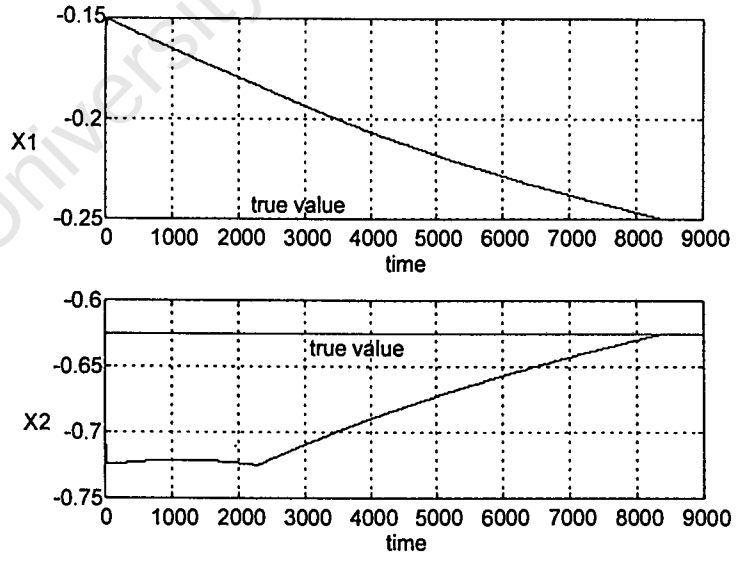

Figure 5.5: *ANN SE with steepest descent dynamic: State estimation as a function of time for the case of parameter error*

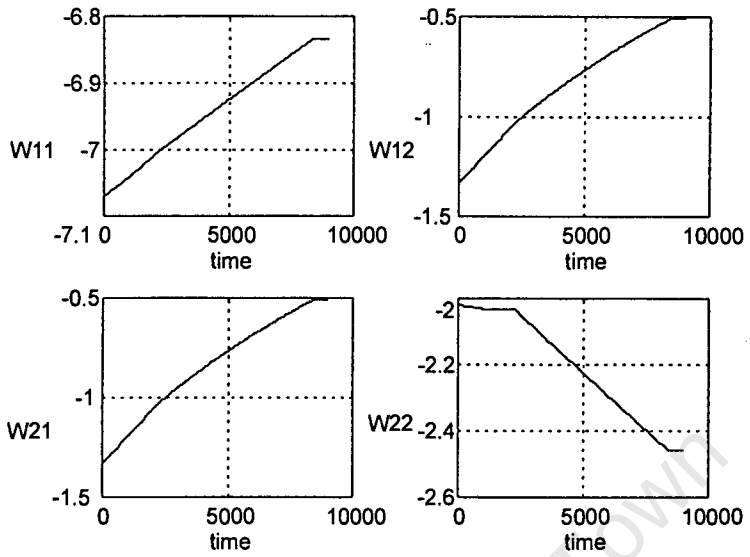

Figure 5.6: *ANN SE with steepest descent dynamic: Weights adaptation as a function of time for the case of parameter error*

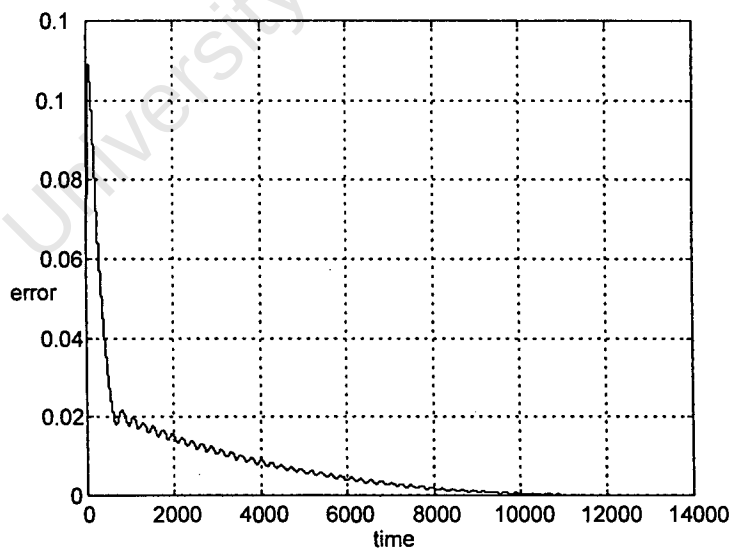

Figure 5.7: *ANN SE with steepest descent dynamic: Estimation error J_e as a function of time for the case of topology error*

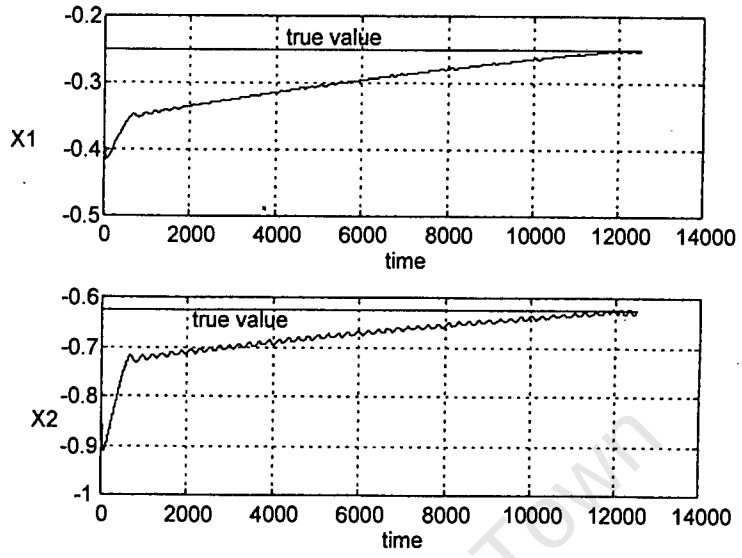

Figure 5.8: *ANN SE with steepest descent dynamic: State estimation as a function of time for the case of topology error*

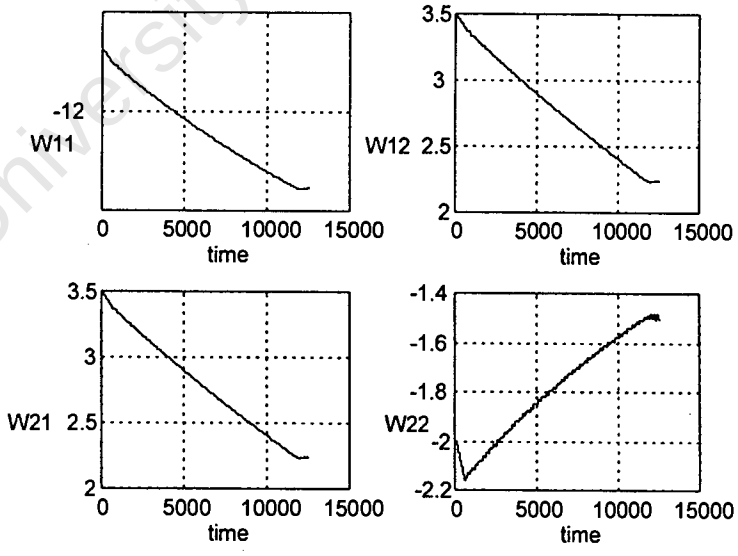

Figure 5.9: *ANN SE with steepest descent dynamic: Weights adaptation as a function of time for the case of topology error*

5.2.2 ANN State Estimator with Hopfield-style Dynamic

In this section the simulation results of the bad data processing technique for the ANN SE given by Eq. (2.71), so called Discrete-Time Recurrent Neural Network (DTRNN), are presented. The bad data processing technique is based on the discrete form of the RTRL (Eqs. (2.72) and (2.73)) where the input T is calculated by using Eq. (5.5).

The first step in this study is the synthesis of the ANN SE, i.e. learning the weights $\mathbf{W}$ of the DTRNN for the initial operating point of the 3-bus test system, specified in Appendix D. Learning was performed by using the RTRL (Eqs. (2.72) and (2.73)) with all weights p equal to 1. The time constants of the DTRNN and RTRL are: $\tau_x = 1$, $\tau_y = 1/0.8$, and $\tau_w = 0.5$. The slope parameter is $c = 1$, and limits are $L_u = -L_l = 1$.

Learning was performed by reducing the sum of squares of the measurement residuals. The learning process is traced by using LS estimation error J_e given by Eq. (5.6), as shown in Figure 5.10. In Figures 5.11 and 5.12 the time evolutions of the estimates towards true state values are presented. Figure 5.13 shows the time evolution of all 4 weights in the process of learning. After learning the ANN state estimator is synthesized for the initial operating point of the 3-bus test system. If the operating point changes the weights need to be adapted. Note, that it is possible to learn DTRNN for more than one operating point as shown in Section 3.3.2. Now the ANN state estimator is ready for testing with bad data.

Gross Measurement Errors

The wrong value for the measurement $z_1 = -0.375$ (instead of 0.375) is presented to the synthetized ANN SE given by Eq. (2.71). Initiated by the measurement error

RTRL automatically changed the set of weights of the ANN SE. At the same time the influence of the bad measurement on state estimates is eliminated.

The parameters of the ANN SE (Eq. (2.71)) stay the same as in the design phase, but the parameters of the RTRL (Eqs. (2.72) and (2.73)) are changed to $\tau_y = 1/0.4$ and $\tau_w = 1/0.1$. The weights p in T (Eq. (5.5)) change according to Eq. (5.4), so that the RTRL minimizes the sum of least absolute values of measurement residuals. The influence of the bad measurement is gradually minimized and the LS estimation error J_e given by Eq. (5.6) is minimized accordingly, as shown in Figure 5.14. The influence of bad measurement z_1 is eliminated by weights adaptation. The time evolutions of the states are shown in Figures 5.15 and 5.16 , and of the weights in Figure 5.17.

Parameter Errors

In this example the error is simulated in the parameter B_{12} (susceptance of the line $1-2$); $B_{12} = 0.1pu$ instead of $1pu$, and the observation matrix $\mathbf{H}$ changed to

$$\mathbf{H} = \begin{pmatrix} 0.1 & -1.5 & 0 & -1.5 & 1.6 \\ -0.1 & 0 & -1 & -1 & -0.1 \end{pmatrix}^T .$$

The original observation matrix is given in Appendix D. The parameter error caused the error in estimation J_e (Eq. (5.6)). The estimation error J_e is eliminated, as shown in Figure 5.18, by using RTRL with the LAV objective function. The parameters of the ANN SE (Eq. (2.71)) stay the same as in the design phase, but the parameters of the RTRL (Eqs. (2.72) and (2.73)) are changed to $\tau_y = 1/0.4$ and $\tau_w = 1/0.1$. The time evolutions of the states are shown in Figures 5.19 and 5.20 , and of the weights in Figure 5.21.

Topology Errors

This example shows how the ANN SE (Eq. (2.71)) with RTRL (Eqs. (2.72) and (2.73)) automatically corrects the estimation errors that occur when the information about line status is incorrect. The incorrect information that line 2 – 3 is not in operation generates the wrong observation matrix

$$\mathbf{H} = \begin{pmatrix} 1 & -1.5 & 0 & -1.5 & 2.5 \\ -1 & 0 & 0 & 0 & -1 \end{pmatrix}^T .$$

The correct one is given in Appendix D.

The RTRL technique with LAV objective function eliminates the estimation error J_e given by Eq. (5.6) caused by topology error, as shown in Figure 5.22. The parameters of the ANN SE stay the same as in the design phase. The parameters of the RTRL are changed to $\tau_y = 1/0.4$ and $\tau_w = 1/0.1$. The time evolutions of the states are shown in Figures 5.23 and 5.24 , and of the weights in the Figure 5.25.

5.2.3 Conclusions

The results of the simulation studies reported in this section lead to the following conclusions:

- the RTRL technique based on LAV objective function, and developed in the form of ANN, is capable of eliminating an estimation error when a gross error in measurement, parameter or topology is present;
- the correction of a state estimate is performed indirectly by adjusting the weights of the ANN SE,
- the method for bad data processing is the same for all types of bad data,

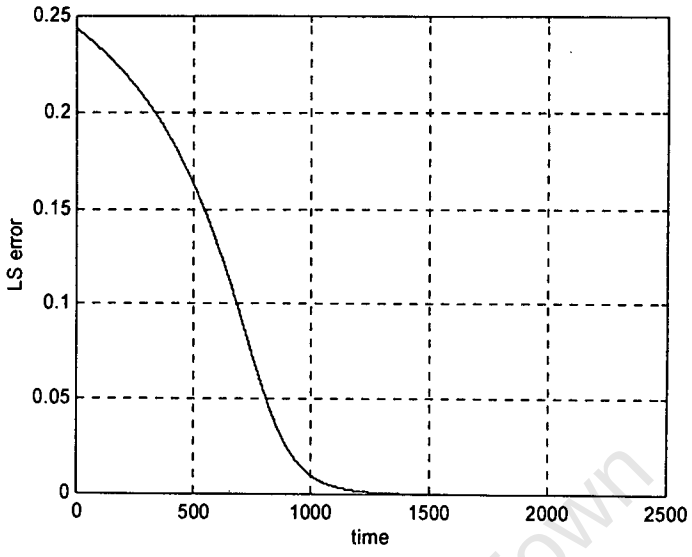

Figure 5.10: *Synthesis of the ANN SE with Hopfield dynamic: Minimization of the LS error J_e in time*

- it is possible to use the RTRL technique for bad data processing in both ANN SE methods: ANN with steepest descent dynamic and ANN with Hopfield dynamic;

- the cases of multiple bad data and combination of different types of bad data need further investigation because the RTRL bad data processing method can in these cases fail in the same way as the classical WLS estimator (with post-processing of measurement residuals) does.

5.3 Properties of Bad Data Processing based on LAV Objective Function

The distinctive feature of LAV minimization applied in ANN SE is that learning of the ANN weights can be obtained by exactly interpolating at least n out of M measurements. It is assumed in this study that the measurement set is observable

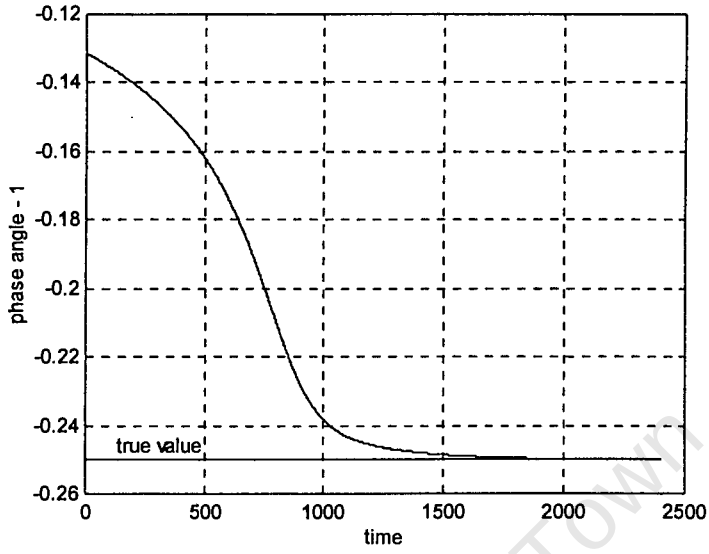

Figure 5.11: *Synthesis of the ANN SE with Hopfield dynamic: Estimate of state x_1 (phase angle 1) as a function of time*

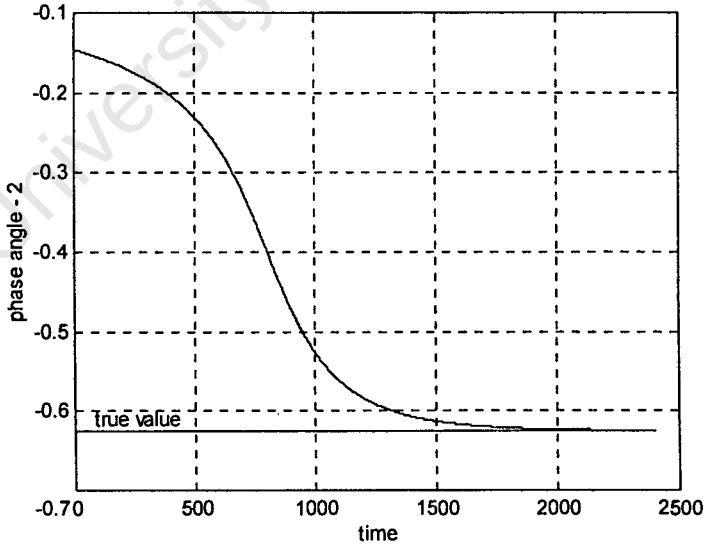

Figure 5.12: *Synthesis of the ANN SE with Hopfield dynamic: Estimate of state x_2 (phase angle 2) as a function of time*

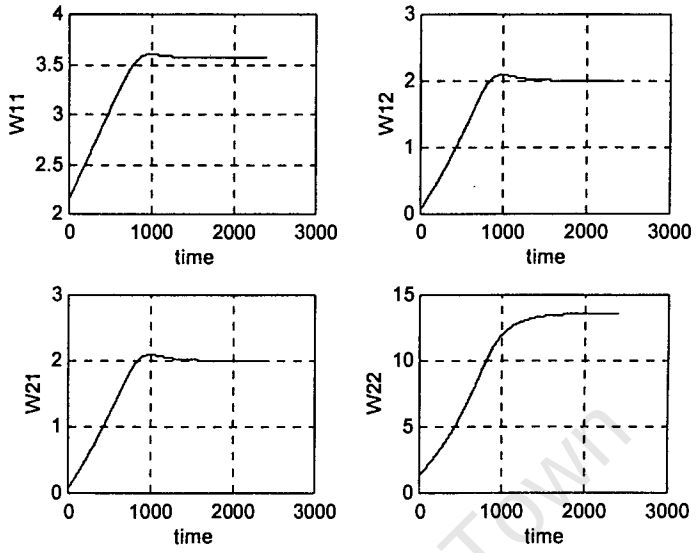

Figure 5.13: *Synthesis of the ANN SE with Hopfield dynamic: Learning the weights in time*

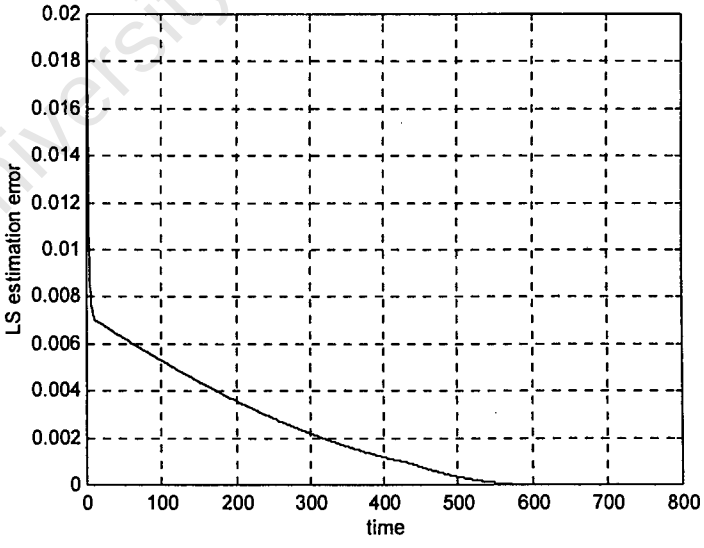

Figure 5.14: *ANN SE with Hopfield dynamic: LS estimation error J_e as a function of time for the case of gross measurement error*

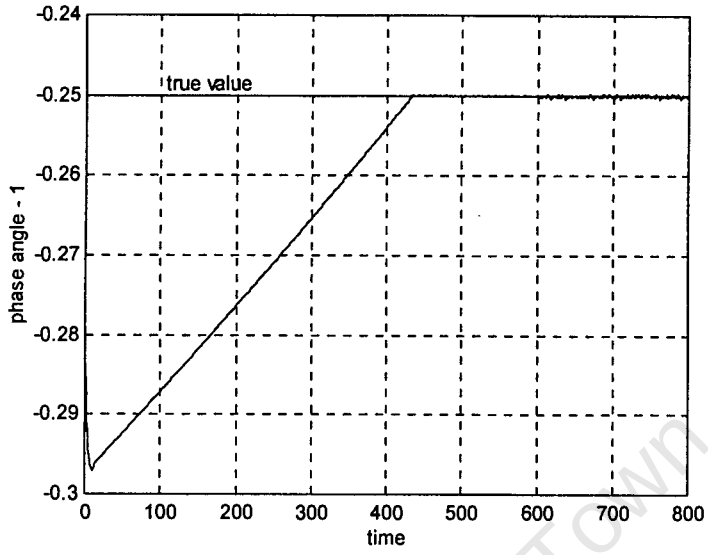

Figure 5.15: *ANN SE with Hopfield dynamic: Estimate of state x_1 (phase angle 1) as a function of time for the case of gross measurement error*

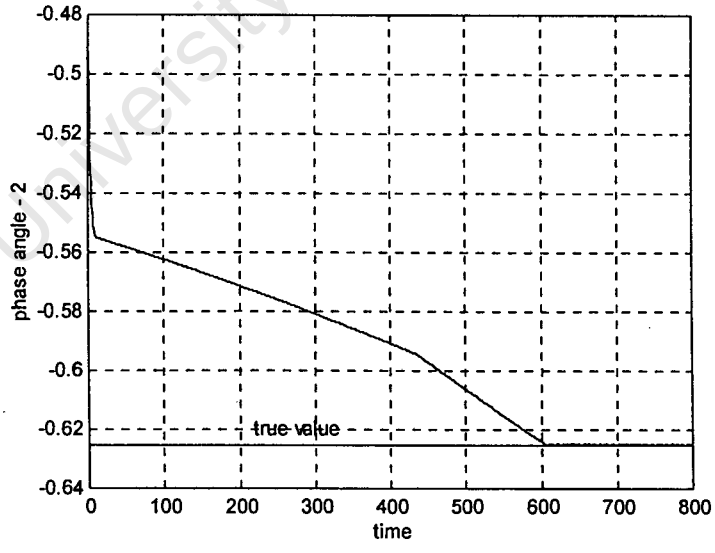

Figure 5.16: *ANN SE with Hopfield dynamic: Estimate of state x_2 (phase angle 2) as a function of time for the case of gross measurement error*

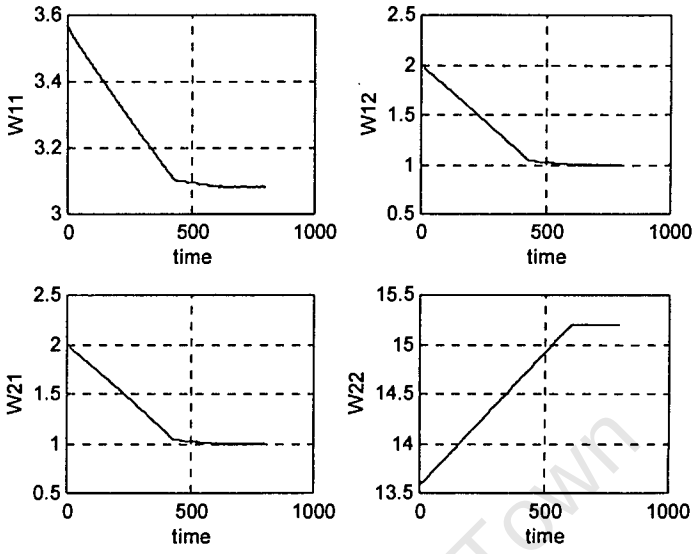

Figure 5.17: *ANN SE with Hopfield dynamic: Weights adaptation in time for the case of gross measurement error*

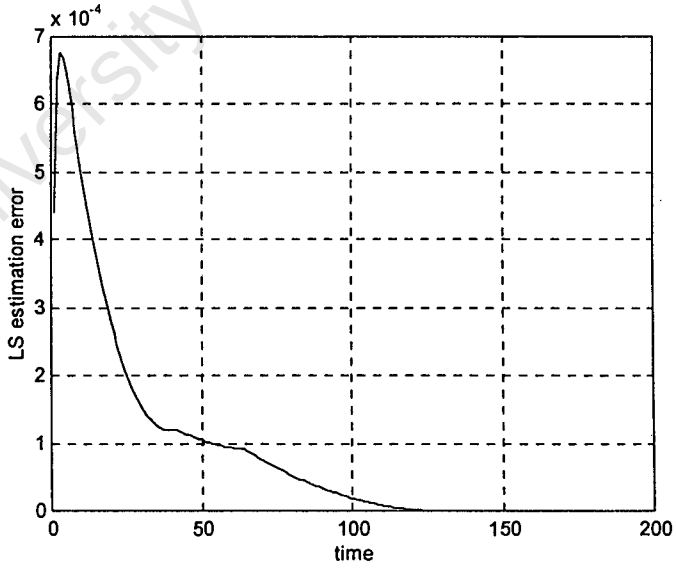

Figure 5.18: *ANN SE with Hopfield dynamic: LS estimation error J_e as a function of time for the case of parameter error*

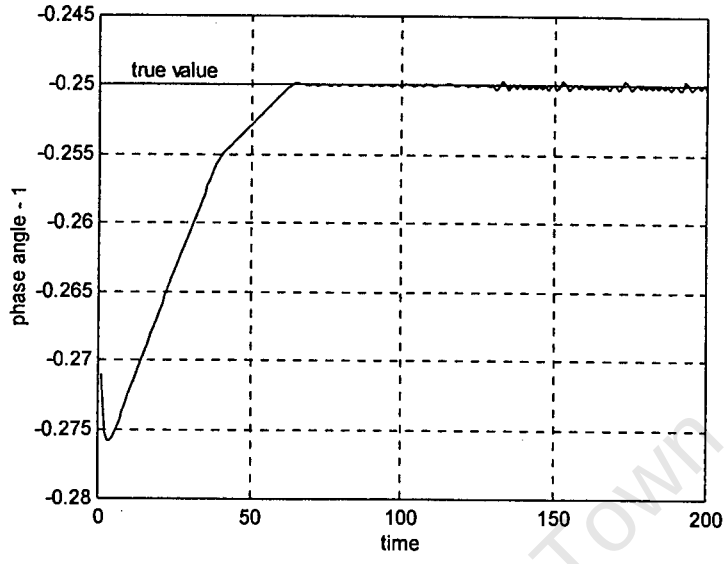

Figure 5.19: *ANN SE with Hopfield dynamic: Estimate of state x_1 (phase angle 1) as a function of time for the case of parameter error*

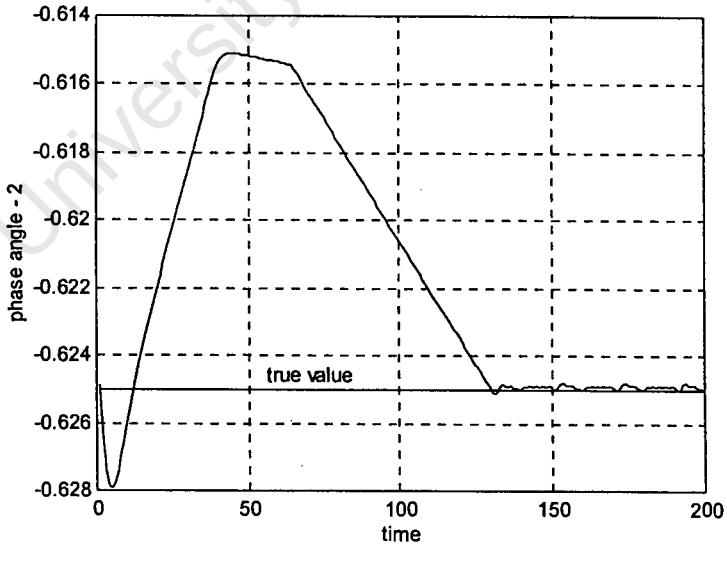

Figure 5.20: *ANN SE with Hopfield dynamic: Estimate of state x_2 (phase angle 2) as a function of time for the case of parameter error*

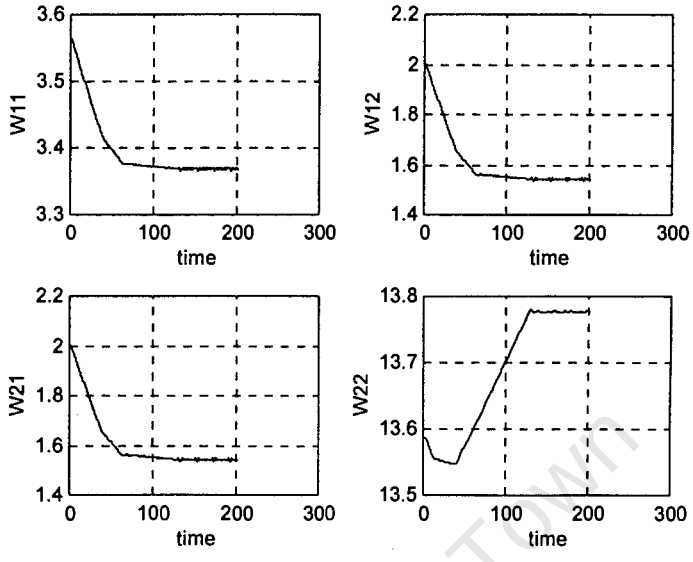

Figure 5.21: *ANN SE with Hopfield dynamic: Weights adaptation in time for the case of parameter error*

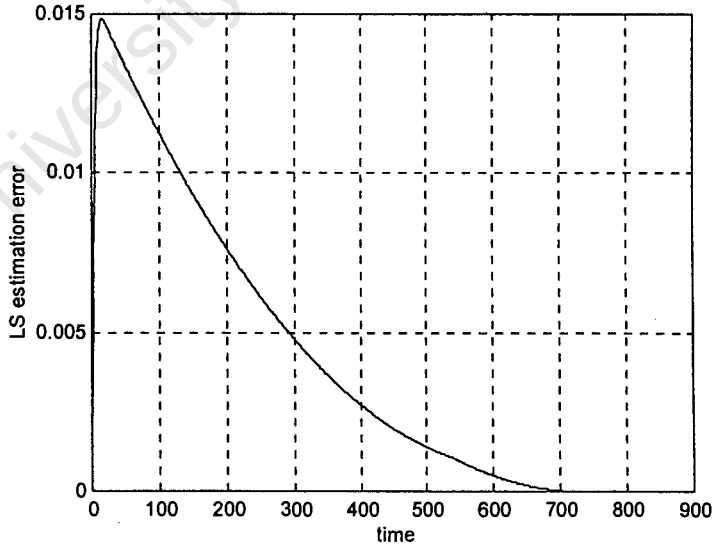

Figure 5.22: *ANN SE with Hopfield dynamic: LS estimation error J_e as a function of time for the case of topology error*

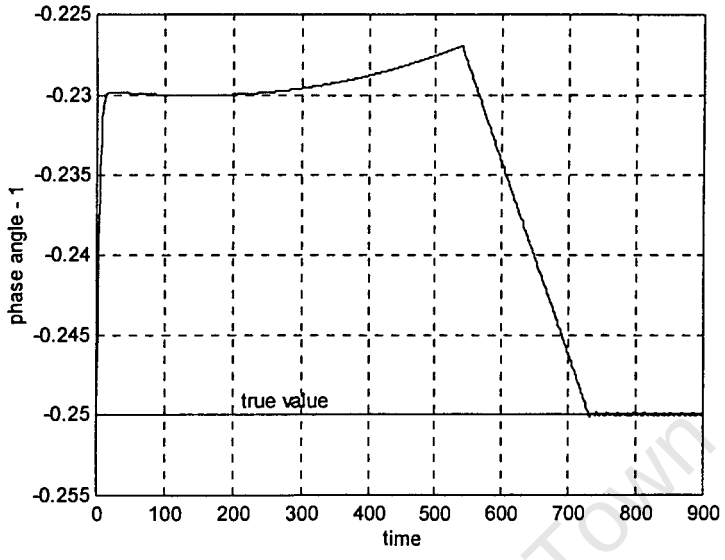

Figure 5.23: *ANN SE with Hopfield dynamic: Estimate of state x_1 (phase angle 1) as a function of time for the case of topology error*

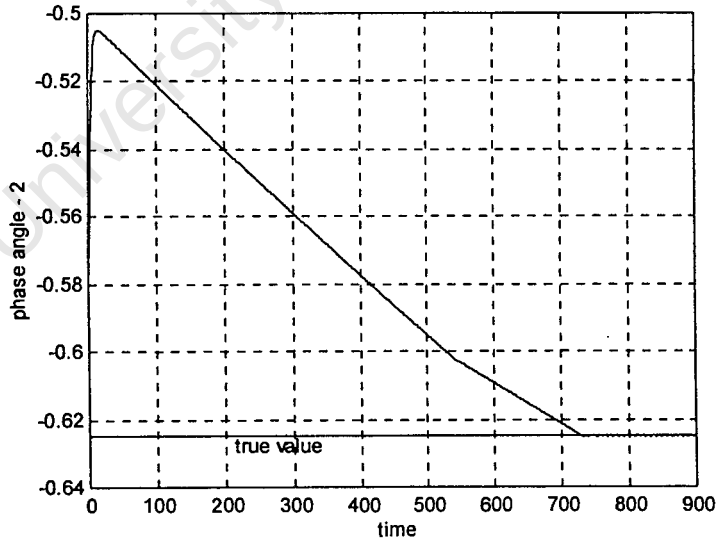

Figure 5.24: *ANN SE with Hopfield dynamic: Estimate of state x_2 (phase angle 2) as a function of time for the case of topology error*

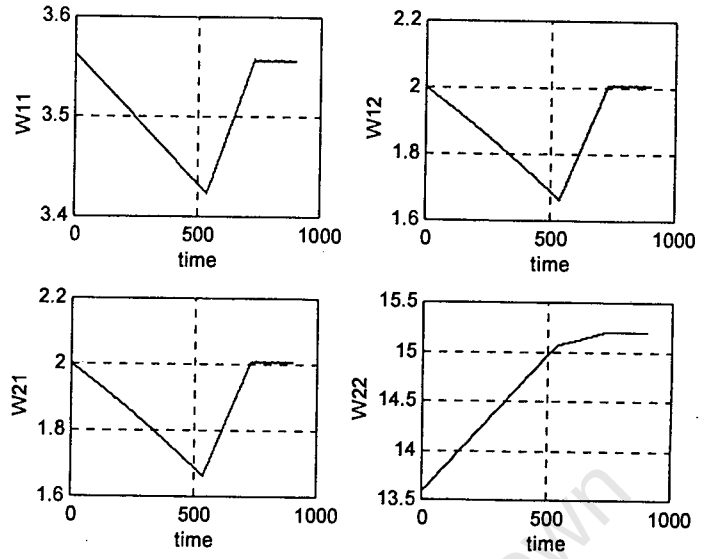

Figure 5.25: *ANN SE with Hopfield dynamic: Weights adaptation in time for the case of topology error*

and that there is sufficient redundancy. The well known ability of LAV minimization to reject bad data without any extra statistical analysis is a direct consequence of the LAV interpolation property.

5.3.1 LAV versus LS Objective Function

In order to derive and to understand the LAV objective function properties, let us linearize and then partition the vector measurement equation (2.1) in the following way [29]

$$\begin{pmatrix} \Delta \mathbf{Z}_t \\ \Delta \mathbf{Z}_s \end{pmatrix} = \begin{pmatrix} \mathbf{H}_t \\ \mathbf{H}_s \end{pmatrix} \Delta \mathbf{X} + \begin{pmatrix} \mathbf{v}_t \\ \mathbf{v}_s \end{pmatrix}, \tag{5.7}$$

and then write the measurement residuals in the form

$$\begin{pmatrix} \mathbf{r}_t \\ \mathbf{r}_s \end{pmatrix} = \begin{pmatrix} \Delta \mathbf{Z}_t \\ \Delta \mathbf{Z}_s \end{pmatrix} - \begin{pmatrix} \mathbf{H}_t \\ \mathbf{H}_s \end{pmatrix} \Delta \hat{\mathbf{X}}, \tag{5.8}$$

where

- t is a subscript and refers to those n measurement equations that are exactly satisfied (residuals equal to 0),
- s is a subscript and refers to the remaining $M - n$ measurement equations suspected of being inaccurate (the reason can be measurement , parameter or topology error), and rejected by LAV.

Substituting Eq. (5.7) into Eq. (5.8) we have

$$\begin{pmatrix} \mathbf{0} \\ \mathbf{r}_s \end{pmatrix} = \begin{pmatrix} \mathbf{H}_t \\ \mathbf{H}_s \end{pmatrix} \left(\Delta\mathbf{X} - \Delta\hat{\mathbf{X}}\right) + \begin{pmatrix} \mathbf{v}_t \\ \mathbf{v}_s \end{pmatrix}. \tag{5.9}$$

After eliminating the state estimation error $\left(\Delta\mathbf{X} - \Delta\hat{\mathbf{X}}\right)$, equation (5.9) can be reduced to the following

$$\mathbf{r}_s = -\mathbf{H}_s\mathbf{H}_t^{-1}\mathbf{v}_t + \mathbf{v}_s. \tag{5.10}$$

Since the mean value of $\mathbf{v}_t$ according to normal distribution is equal to $\mathsf{E}[\mathbf{v}_t] = 0$, the expected residual vector $\mathbf{r}_s$ of the suspected measurement equations corresponds directly to error $\mathbf{v}_s$.

When the LS objective function is used to estimate $\hat{\mathbf{X}}$ from a linearized measurement (Eq. (2.1)) the measurement residual vector is (according to derivation in Appendix B)

$$\bar{\mathbf{r}} = \Delta\mathbf{Z} - \mathbf{H}\Delta\hat{\mathbf{X}} = \mathcal{W}\mathbf{v}, \tag{5.11}$$

where $\mathcal{W}$ is the $M \times M$ residual sensitivity matrix. These residuals are used in bad data detection and identification procedure in the classical WLS State Estimator [24]. Using the same notation as in Eq. (5.8), equation (5.11) can be partitioned as follows

$$\begin{pmatrix} \bar{\mathbf{r}}_t \\ \bar{\mathbf{r}}_s \end{pmatrix} = \begin{pmatrix} \mathcal{W}_{tt} & \mathcal{W}_{ts} \\ \mathcal{W}_{st} & \mathcal{W}_{ss} \end{pmatrix} \begin{pmatrix} \mathbf{v}_t \\ \mathbf{v}_s \end{pmatrix}. \tag{5.12}$$

In the LS case the subscript s refers to the maximum number $(M-n)$ of suspected data (from detection test [24]), and the subscript t indicates an $n \times 1$ vector containing the smallest residuals. We can write the lower part of the equation (5.12) as

$$\bar{\mathbf{r}}_s = \mathcal{W}_{ss}\mathbf{v}_s + \mathcal{W}_{st}\mathbf{v}_t. \tag{5.13}$$

Error $\mathbf{v}_s$ in the suspected measurement equations can be estimated (bad data identification phase [24]) from $\bar{\mathbf{r}}_s$, by using the LS approach

$$\hat{\mathbf{v}}_s = \mathcal{W}_{ss}^{-1}\bar{\mathbf{r}}_s. \tag{5.14}$$

Here we assumed that $\mathbf{v}_t$ is normally distributed with a zero mean value.

The corrected estimation vector $\Delta\hat{\mathbf{X}}^c$ can be obtained using the corrected measurement vector $\begin{pmatrix} \Delta\mathbf{Z}_t & \Delta\mathbf{Z}_s - \hat{\mathbf{v}}_s \end{pmatrix}^T$. As the residuals of the corrected measurements are equal to zero (precisely the mean of these residuals because $\mathbf{v}_t$ is assumed to be normally distributed with zero mean) we have

$$\bar{\mathbf{r}}_s^c = \mathcal{W}_{ss}\left(\mathbf{v}_s - \hat{\mathbf{v}}_s\right) + \mathcal{W}_{st}\mathbf{v}_t = \mathbf{0}. \tag{5.15}$$

This means that the corrected vector $\Delta\hat{\mathbf{X}}^c$ is obtained through $\Delta\mathbf{Z}_t$ only.

Further we can prove that the vector of error estimates $\hat{\mathbf{v}}_s$ is the same as the vector of residuals $\mathbf{r}_s$ in the LAV approach. Substituting Eq. (5.13) into Eq. (5.14) we obtain equation (5.10) (according to derivation in Appendix B)

$$\hat{\mathbf{v}}_s = \mathcal{W}_{ss}^{-1}\mathcal{W}_{st}\mathbf{v}_t + \mathbf{v}_s = -\mathbf{H}_s\mathbf{H}_t^{-1}\mathbf{v}_t + \mathbf{v}_s = \mathbf{r}_s. \tag{5.16}$$

The statistical properties of the corrected state estimation error $\left(\Delta\mathbf{X} - \Delta\hat{\mathbf{X}}^c\right)$ and the error estimates $\hat{\mathbf{v}}_s$ are the same as the statistical properties of estimation error and residuals $\mathbf{r}_s$ in LAV, respectively.

In conclusion: the bad data processing based on the LAV objective function gives the same result as classical bad data detection and identification method based on the post residual analysis in the LS approach [24].

5.3.2 Problems with the LAV Objective Function

From equation (5.10) it follows that the source of error could be $\mathbf{v}_s$ and/or $\mathbf{v}_t$. The following situations will be considered:

a) The bad data (measurements and/or parameters) belonging to subset s in equation (5.8); the state estimate is properly corrected through the RTRL with the LAV objective function.

b) The bad data belonging to subset t; RTRL with the LAV objective can not give accurate results.

c) The bad data belonging to both subsets; RTRL can not correct the error in the state estimate.

In situations b) and c) the RTRL with LAV objective function always fails in eliminating the influence of bad data on the state estimate. These cases are rare, however they are still possible, and further research is therefore necessary.

Cases b) and c) result from the critical measurements, multiple interacting bad data, and/or bad leverage points in the state estimation regression model given by Eq. (2.1). The critical measurement is the one whose deletion will cause the system to become unobservable. The leverage points are those points which are far away from the bulk of the data points in the factor space [33]. In the regression model of the power system state estimation, the factor space is represented by the Jacobian matrix $\mathbf{H}$. In a power system, the leverage points result from measurements of the following types [33]: line flows and bus injections associated with lines that are relatively short compared to the others and bus injections at nodes that have a large number of incident lines. The entries of the Jacobian matrix $\mathbf{H}$ associated with such measurements become large. Whether the leverage point is good or bad, depends upon whether the metered value, which corresponds to the leverage point, is valid

or not. A wrong parameter associated with a valid metered value can also create a bad leverage point.

Detection of bad data in situations b) and c) may be achieved, in some cases, by utilizing the linear regression model (5.10), as shown in [30]. The gross errors in subset t can be identified directly as long as the subset s is cleared of bad data (as in case b)). In estimating $\mathbf{v}_t$ the model (5.10) is assumed, and the LS algorithm is applied

$$\hat{\mathbf{v}}_t = \left(\mathcal{H}^T \mathbf{R}_s^{-1} \mathcal{H}\right)^{-1} \mathcal{H}^T \mathbf{R}_s^{-1} \mathbf{r}_s, \tag{5.17}$$

where $\mathcal{H} = -\mathbf{H}_s \mathbf{H}_t^{-1}$ and $\mathbf{R}_s^{-1}$ is the inverse of the covariance matrix of measurement errors for subset s. In the situation c) the subset s is contaminated with bad data, so the regression model (5.10) is biased. The masking effect of bad data in subset s makes the identification of any gross errors in subset t unreliable. However, we can use a hypothesis test based on the following residual

$$\mathbf{r}_p = \mathbf{r}_s - \mathcal{H}\hat{\mathbf{v}}_t, \tag{5.18}$$

in weighted or normalized form [30]. Sometimes this technique can not succeed due to rank deficiency of matrix $\mathcal{H}$ and/or bad leverage points in the factor space of the regression model (5.10), as shown in [30].

Chapter 6

Conclusions

This thesis presents a new approach in solving the power system state estimation problem. This approach is based on the latest developments in the area of Artificial Neural Networks (ANN). In the thesis a number of power system state estimation methods are formulated as ANN, therefore we introduced the name ANN State Estimator (ANN SE) for this new class of estimators. It has been shown that the ANN SE has a better filtering performance compared to the traditional WLS method. Implementation by using an ANN hardware can speed up the state estimation function in a power system control centre.

The new bad data processing method has been developed specially for the ANN SE. The base for this method is the Real Time Recurrent Learning (RTRL) algorithm. The idea is to change the ANN SE parameters once a bad data is presented to the estimator. In this manner the ANN SE output is automatically corrected, and the influence of bad data is completely rejected. It has been shown that the method is capable to efficiently cope with different types of bad data: bad measurements, bad parameters or bad topology. This bad data processing method can be implemented on an ANN hardware, which will speed up the computation. In continuation bad data can be identified by using the existing techniques discussed in section 1.2.

In general, this thesis presents the foundation of the new ANN SE concept, which can bring certain improvements over the traditional power system state estimation methods.

6.1 Contributions of the Thesis

This thesis makes several contributions to the area of power system state estimation. The contributions are as follows:

- Section 2.3: Fundamentals of the ANN SE with the Steepest Descent Dynamic
  The power system state estimation problem is formulated as the optimization problem. We used the ANN with the steepest descent dynamic as the solver for that optimization problem. In order to simplify the implementation we have done the following: zero injection measurements are incorporated in the ANN SE through the variable reduction technique, thus decreasing the number of neurons; and the special algebraic transformation of the ANN SE objective function is introduced decreasing the number of operations necessary to compute the ANN SE parameters. It is shown that the hyperbolic-tangent activation function improves filtering of small measurement errors. The formulas for calculating the sensitivity measure of the ANN SE (with a specific structure and parameters) output to the input perturbations (white noise) are developed. Two methods for designing the ANN SE with steepest descent dynamic for a specific power system are proposed. The first method (so called analytic) maps the structure and the parameters of a power system into the ANN SE structure and parameters. The second method (so called synthetic) employes the Real Time Recurrent Learning (RTRL) algorithm to determine weights of the ANN SE from the available input/output data. Examples given in this section illustrate the new developments.

- Section 2.4: Fundamentals of the ANN SE with the Hopfield-style Dynamic

  In this case the solver for the state estimation problem is the ANN with the Hopfield-style dynamic. Two types of the ANN with Hopfield-style dynamic are used in the development of the ANN SE: the Hopfield ANN and the Continuous-Time Recurrent Neural Network (CTRNN). Analytic approach in design is not possible in this case. The synthetic design method for this class of the ANN SE is developed using the RTRL technique. The sensitivity analysis method is developed to determine the performance of the ANN SE with Hopfield-style dynamic. Examples given in this section illustrate the new developments.

- Chapter 3: Development of the feasible ANN SE methods

  The ANN SE methods presented in Chapter 2 are not feasible for large-scale power systems so we introduced the feasible ANN SE by doing the following two simplifications: decoupling $P\theta$ loop from QV loop, and making ANN weight matrix constant. Both classes of the ANN SE (the steepest descent and the Hopfield-style) are simplified and called by common name: Decoupled ANN SE. We showed how to use the Decoupled ANN SE in static and tracking state estimation. Two versions of Decoupled ANN SE are tested through sensitivity and simulation studies on the IEEE 14-bus test system. The results are presented in this chapter.

- Chapter 4: Use of the voltage phasor angle measurements in the ANN SE

  ANN SE is formulated in such a way that the voltage phasor angle measurements can be included. Two types of angle measurements are considered: absolute bus voltage phasor angle measurements, and angle difference between voltage phasor measurements in adjacent buses. It is shown that angle difference has an advantage because the angle reference problem does not exist.

Simulation and sensitivity studies on the IEEE 14-bus test power system show the benefits of using angle measurements. However, the conclusion is that the angle measurements should only complement (not replace) power flow and injection measurements.

- Chapter 5: Fundamentals of the bad data processing technique for the ANN SE

  In this chapter we developed the bad data processing technique based on the RTRL with the Least Absolute Value (LAV) objective function. The technique automatically adjusts the parameters of the ANN SE, eliminating the influence of bad data on the estimator output. It works equally well on the measurement, parameter, or topology errors, as shown in the presented examples.

6.2 ANN State Estimator versus LS and LAV Estimators

Two issues, one, the accuracy of the estimates and two, the speed of estimation are of interest to the state estimator user. Therefore, we would like to summarize the benefits observed in these two areas due to the use of ANN based estimator versus the existing LS and LAV estimators. The benefits are as follows:

- Section 2.3: Improving state estimation accuracy

  The hyperbolic-tangent activation function used in ANN SE improves filtering of Gaussian noise compared to LS and LAV estimators.

- Section 2.3 and 2.4: Improving state estimation accuracy

  The existing LS estimator is an optimal filter only for Gaussian noise, while for other noise types it has poor performance. The synthetic design of ANN SE does not require Gaussian assumption and therefore allows better filtering for unknown error distribution.

- Chapter 5: Improving state estimation robustness

  ANN SE does uniform processing of all kinds of bad data, while LS estimator needs always additional post-processing and LAV estimator needs post-processing for parameter and topological errors.

- Section 3.3.1: Speeding of state estimation

  At present the duration of a state estimation based on the existing LS or LAV algorithms is longer than the SCADA scan cycle (few seconds). It has been shown in this section that use of ANN SE on analog ANN hardware would produce an estimate for large power systems in few ms.

6.3 Future Work

The following are suggestions for future work:

- Develop a bad data processing technique for the ANN SE based on the robust estimation theory. This technique should use for the RTRL objective, either the minimization of the median of the squared residuals or, the minimization of the sum of the smallest ordered squared residuals up to certain rank v ($v < M$). Such a technique will be able to cope with multiple bad leverage points.

- Implementation of the ANN SE and the bad data processing technique on a special purpose ANN hardware. The hardware can be based either on a DSP processor or, on an analog ANN processor. This will make possible real-time application of the ANN SE in medium and large scale power networks.

Appendix A

Power System Model for State Estimation

In this appendix, we present the network equations, which are used to compute $\mathbf{h}(\mathbf{X})$. Then we derive equations for the elements of the Jacobian matrix $\mathbf{H}(\mathbf{X}) = \frac{\partial \mathbf{h}(\mathbf{X})}{\partial \mathbf{X}}$. Finally we simplify the Jacobian matrix and make $\mathbf{H}$ constant (independent of a system state).

A power transmission line is often modeled as the π-circuit [72]. Active power flow (Pf_{ij}) and reactive power flow (Qf_{ij}) at node i toward node j are given by

$$Pf_{ij} = G_{ij}V_i^2 - V_iV_j\left(G_{ij}\cos\theta_{ij} + B_{ij}\sin\theta_{ij}\right) \tag{A.1}$$

$$Qf_{ij} = -B'_{ij}V_i^2 - V_iV_j\left(G_{ij}\sin\theta_{ij} - B_{ij}\cos\theta_{ij}\right) \tag{A.2}$$

where

$$\theta_{ij} = \theta_i - \theta_j, \tag{A.3}$$

and

V_i, θ_i - voltage magnitude and phase angle at node i,

V_j, θ_j - voltage magnitude and phase angle at node j,

G_{ij} - $(i-j)$ element of the real part of bus admittance matrix $\mathbf{Y}$,

B_{ij} - $(i-j)$ element of the imaginary part of $\mathbf{Y}$,

$B'_{ij} = B_{ij} + b_{ij}/2$,

b_{ij} - susceptance (shunt capacitance) of line $(i-j)$.

Expressions for G_{ij} and B_{ij} as functions of transmission line parameters R_{ij} (resistance) and X_{ij} (reactance), are

$$G_{ij} = \frac{R_{ij}}{R_{ij}^2 + X_{ij}^2}, \quad B_{ij} = -\frac{X_{ij}}{R_{ij}^2 + X_{ij}^2}. \tag{A.4}$$

Let P_i and Q_i respectively denote active and reactive power injections at node i. Then by using the conservation law and Eqs. (A.1) and (A.2), we obtain

$$P_i = \sum_{j \in \beta(i)} Pf_{ij} = G_{ii}V_i^2 - V_i \sum_{j \in \beta(i)} V_j \left(G_{ij} \cos\theta_{ij} + B_{ij} \sin\theta_{ij}\right), \tag{A.5}$$

$$Q_i = \sum_{j \in \beta(i)} Qf_{ij} = -B_{ii}V_i^2 - V_i \sum_{j \in \beta(i)} V_j \left(G_{ij} \sin\theta_{ij} - B_{ij} \cos\theta_{ij}\right), \tag{A.6}$$

where

$$G_{ii} = \sum_{j \in \beta(i)} G_{ij}, \quad B_{ii} = \sum_{j \in \beta(i)} B_{ij} \tag{A.7}$$

and $\beta(i)$ denotes the set of all indices of the nodes directly connected to node i. The above equations are used to construct the vector function $\mathbf{h}(\mathbf{X})$ given in Eqs. (2.1) and (2.5). It is possible to divide this vector function according to active and reactive sets of measurements ($\mathbf{Z}_P$ and $\mathbf{Z}_Q$) into two subvectors

$$\mathbf{h}(\mathbf{X}) = \begin{pmatrix} \mathbf{h}_P(\Theta, \mathbf{V}) \\ \mathbf{h}_Q(\Theta, \mathbf{V}) \end{pmatrix}, \tag{A.8}$$

where

$$\mathbf{h}_P(\Theta, \mathbf{V}) = \begin{pmatrix} \mathbf{P}(\Theta, \mathbf{V}) \\ \mathbf{Pf}(\Theta, \mathbf{V}) \\ \Theta \end{pmatrix}, \quad \mathbf{h}_Q(\Theta, \mathbf{V}) = \begin{pmatrix} \mathbf{Q}(\Theta, \mathbf{V}) \\ \mathbf{Qf}(\Theta, \mathbf{V}) \\ \mathbf{V} \end{pmatrix}. \tag{A.9}$$

In the following, we list the formulas for various derivatives needed in computing the Jacobian $\mathbf{H}(\mathbf{X}) = \frac{\partial \mathbf{h}(\mathbf{X})}{\partial \mathbf{X}}$.

1.

$$\frac{\partial P_i}{\partial \theta_r} = \begin{cases} V_i \sum_{j \in \beta(i)} V_j \left(G_{ij} \sin\theta_{ij} - B_{ij} \cos\theta_{ij}\right) = -Q_i - B_{ii} V_i^2 & \text{for } r = i, \\ -V_i V_r \left(G_{ir} \sin\theta_{ir} - B_{ir} \cos\theta_{ir}\right) & \text{for } r \in \beta(i), \\ 0 & \text{otherwise.} \end{cases} \tag{A.10}$$

2.

$$\frac{\partial P_i}{\partial V_r} = \begin{cases} 2G_{ii} V_i - \sum_{j \in \beta(i)} V_j \left(G_{ij} \cos\theta_{ij} + B_{ij} \sin\theta_{ij}\right) = \frac{P_i}{V_i} + G_{ii} V_i & \text{for } r = i, \\ -V_i \left(G_{ir} \cos\theta_{ir} + B_{ir} \sin\theta_{ir}\right) & \text{for } r \in \beta(i), \\ 0 & \text{otherwise.} \end{cases} \tag{A.11}$$

3.

$$\frac{\partial Q_i}{\partial \theta_r} = \begin{cases} -V_i \sum_{j \in \beta(i)} V_j \left(G_{ij} \cos\theta_{ij} + B_{ij} \sin\theta_{ij}\right) = P_i - G_{ii} V_i^2 & \text{for } r = i, \\ V_i V_r \left(G_{ir} \cos\theta_{ir} + B_{ir} \sin\theta_{ir}\right) & \text{for } r \in \beta(i), \\ 0 & \text{otherwise.} \end{cases} \tag{A.12}$$

4.

$$\frac{\partial Q_i}{\partial V_r} = \begin{cases} -2B_{ii} V_i - \sum_{j \in \beta(i)} V_j \left(G_{ij} \sin\theta_{ij} - B_{ij} \cos\theta_{ij}\right) = \frac{Q_i}{V_i} - B_{ii} V_i & \text{for } r = i, \\ -V_i \left(G_{ir} \sin\theta_{ir} - B_{ir} \cos\theta_{ir}\right) & \text{for } r \in \beta(i), \\ 0 & \text{otherwise.} \end{cases} \tag{A.13}$$

5.

$$\frac{\partial Pf_{ij}}{\partial \theta_r} = \begin{cases} V_i V_j \left(G_{ij} \sin\theta_{ij} - B_{ij} \cos\theta_{ij}\right) & \text{for } r = i, \\ -V_i V_j \left(G_{ij} \sin\theta_{ij} - B_{ij} \cos\theta_{ij}\right) & \text{for } r = j, \\ 0 & \text{otherwise.} \end{cases} \tag{A.14}$$

6.

$$\frac{\partial Pf_{ij}}{\partial V_r} = \begin{cases} 2G_{ij}V_i - V_j\left(G_{ij}\cos\theta_{ij} + B_{ij}\sin\theta_{ij}\right) & \text{for } r = i, \\ -V_i\left(G_{ij}\cos\theta_{ij} + B_{ij}\sin\theta_{ij}\right) & \text{for } r = j, \\ 0 & \text{otherwise.} \end{cases} \tag{A.15}$$

7.

$$\frac{\partial Qf_{ij}}{\partial \theta_r} = \begin{cases} -V_iV_j\left(G_{ij}\cos\theta_{ij} + B_{ij}\sin\theta_{ij}\right) & \text{for } r = i, \\ V_iV_j\left(G_{ij}\cos\theta_{ij} + B_{ij}\sin\theta_{ij}\right) & \text{for } r = j, \\ 0 & \text{otherwise.} \end{cases} \tag{A.16}$$

8.

$$\frac{\partial Qf_{ij}}{\partial V_r} = \begin{cases} -2B'_{ij}V_i - V_j\left(G_{ij}\sin\theta_{ij} - B_{ij}\cos\theta_{ij}\right) & \text{for } r = i, \\ -V_i\left(G_{ij}\sin\theta_{ij} - B_{ij}\cos\theta_{ij}\right) & \text{for } r = j, \\ 0 & \text{otherwise.} \end{cases}$$

9.

$$\frac{\partial V_i}{\partial \theta_r} = 0, \tag{A.17}$$

10.

$$\frac{\partial V_i}{\partial V_r} = \begin{cases} 1 & \text{for } r = i, \\ 0 & \text{otherwise.} \end{cases}, \tag{A.18}$$

11.

$$\frac{\partial \theta_{ij}}{\partial V_r} = 0, \tag{A.19}$$

12.

$$\frac{\partial \theta_{ij}}{\partial \theta_r} = \begin{cases} 1 & \text{for } r = i, \\ -1 & \text{for } r = j, \\ 0 & \text{otherwise.} \end{cases} \tag{A.20}$$

Finally the Jacobian matrix for the full estimator is

$$\mathbf{H}(\mathbf{V}, \Theta) = \begin{pmatrix} \mathbf{H}_{P\theta} & \mathbf{H}_{PV} \\ \mathbf{H}_{Q\theta} & \mathbf{H}_{QV} \end{pmatrix} \tag{A.21}$$

where

$$\begin{aligned} \mathbf{H}_{P\theta} &= \frac{\partial \mathbf{h}_P(\Theta, \mathbf{V})}{\partial \Theta}, \qquad \mathbf{H}_{PV} = \frac{\partial \mathbf{h}_P(\Theta, \mathbf{V})}{\partial \mathbf{V}}, \\ \mathbf{H}_{Q\theta} &= \frac{\partial \mathbf{h}_Q(\Theta, \mathbf{V})}{\partial \Theta}, \qquad \mathbf{H}_{QV} = \frac{\partial \mathbf{h}_Q(\Theta, \mathbf{V})}{\partial \mathbf{V}}. \end{aligned} \tag{A.22}$$

The sensitivity of active measurements $\mathbf{Z}_P$ to changes in voltage magnitudes is small, and we assume $\mathbf{H}_{PV} = 0$. The same is valid for the sensitivity of reactive measurements $\mathbf{Z}_Q$ on changes in voltage phase angle, and $\mathbf{H}_{Q\theta} = 0$. This property is exploited in the development of a decoupled state estimator. Furthermore, based on physical characteristics of the power networks the elements of matrices $\mathbf{H}_{P\theta}$ and $\mathbf{H}_{QV}$ can be made state independent by using the following approximations

$$\begin{aligned} \cos\theta_{ij} &\simeq 1, \\ V_i &\simeq 1, \\ G_{ij}\sin\theta_{ij} &\ll B_{ij}\cos\theta_{ij}. \end{aligned} \tag{A.23}$$

Introducing approximations (A.23), the elements of $\mathbf{H}_{P\theta}$ and $\mathbf{H}_{QV}$ are given by the equations

$$\frac{\partial P_i}{\partial \theta_r} = \begin{cases} -B_{ii} & \text{for } r = i \\ B_{ir} & \text{for } r \in \beta(i) \\ 0 & \text{otherwise.} \end{cases}, \qquad \frac{\partial Pf_{ij}}{\partial \theta_r} = \begin{cases} -B_{ij} & \text{for } r = i \\ B_{ij} & \text{for } r = j \\ 0 & \text{otherwise.} \end{cases}, \tag{A.24}$$

$$\frac{\partial Q_i}{\partial V_r} = \begin{cases} -B_{ii} & \text{for } r = i \\ B_{ir} & \text{for } r \in \beta(i) \\ 0 & \text{otherwise.} \end{cases}, \qquad \frac{\partial Qf_{ij}}{\partial V_r} = \begin{cases} -2B'_{ij} + B_{ij} & \text{for } r = i \\ B_{ij} & \text{for } r = j \\ 0 & \text{otherwise.} \end{cases}.$$

Appendix B

Sensitivity Analysis

B.1 Static State Estimation

In the case of static ANN SE the sensitivity analysis makes use of the linearization of $\mathbf{h}(\mathbf{X})$ around $\hat{\mathbf{X}}$ given by

$$\mathbf{h}(\mathbf{X}) \simeq \mathbf{h}\left(\hat{\mathbf{X}}\right) + \mathbf{H}\left(\hat{\mathbf{X}}\right)\left(\mathbf{X} - \hat{\mathbf{X}}\right). \tag{B.1}$$

Three kinds of sensitivity parameters are presented:

- Sensitivities relating small measurement errors to state estimates. These parameters can be used to study effects of small measurement errors on estimation accuracy.
- Sensitivities relating small measurement errors to difference between true measurement and estimated measurement values.
- Sensitivities relating gross measurement errors to measurement residuals (difference between measured and estimated measurement values). They are used in detection and identification of gross measurement errors, and the estimation of parameter errors.

When the convergence of an ANN SE is achieved, one has $\mathbf{X}_{k+1} \approx \mathbf{X}_k = \hat{\mathbf{X}}$, and the gradient of the objective function is $\nabla_x J = 0$. Now, from Eq. (2.17) we find

$$\mathbf{H}^T\left(\hat{\mathbf{X}}\right)\left[\mathbf{Z} - \mathbf{h}\left(\hat{\mathbf{X}}\right)\right] = \mathbf{0}. \tag{B.2}$$

In the above equation, and in all equations which follow, we use the transformed Jacobian matrix $\mathbf{H}\left(\hat{\mathbf{X}}\right) \equiv \mathbf{R}^{-1/2}\mathbf{H}\left(\hat{\mathbf{X}}\right)$; therefore it is not necessary to explicitly consider the covariance matrix $\mathbf{R}$ given by Eq. (2.2) in these equations. Substituting equations (2.1) and (B.1) into Eq. (B.2) yields

$$\mathbf{H}^T\left(\hat{\mathbf{X}}\right)\left[\mathbf{h}\left(\mathbf{X}\right)+\mathbf{v}-\mathbf{h}\left(\mathbf{X}\right)+\mathbf{H}\left(\hat{\mathbf{X}}\right)\left(\mathbf{X}^*-\hat{\mathbf{X}}\right)\right]=\mathbf{0},$$

or equivalently,

$$\mathbf{X}^*-\hat{\mathbf{X}}=-\left[\mathbf{H}^T\left(\hat{\mathbf{X}}\right)\mathbf{H}\left(\hat{\mathbf{X}}\right)\right]^{-1}\mathbf{H}^T\left(\hat{\mathbf{X}}\right)\mathbf{v} \tag{B.3}$$

Hence,

$$\mathbf{X}^*-\hat{\mathbf{X}}=-\mathbf{P}\mathbf{v}, \tag{B.4}$$

where

$$\mathbf{P}=\left[\mathbf{H}^T\left(\hat{\mathbf{X}}\right)\mathbf{H}\left(\hat{\mathbf{X}}\right)\right]^{-1}\mathbf{H}^T\left(\hat{\mathbf{X}}\right) \tag{B.5}$$

is the $n \times n$ sensitivity matrix. If the small measurement errors $\mathbf{v}$ are normally distributed with zero mean value, then according to Eq. (B.4) the estimation errors $\mathbf{X}^*-\hat{\mathbf{X}}$ are normally distributed with a zero mean and covariance matrix

$$\mathrm{E}\left[\left(\mathbf{X}^*-\hat{\mathbf{X}}\right)\left(\mathbf{X}^*-\hat{\mathbf{X}}\right)^T\right]=\left[\mathbf{H}^T\left(\hat{\mathbf{X}}\right)\mathbf{H}\left(\hat{\mathbf{X}}\right)\right]^{-1}. \tag{B.6}$$

This covariance matrix provides an indication of the expected accuracy of the state estimate. The sensitivity matrix (B.5) can also be used to assess the effect of measurement bias on the state estimate.

Substituting equations (2.1) and (B.1) in the measurement residual equation yields

$$\mathbf{r}=\mathbf{Z}-\mathbf{h}\left(\hat{\mathbf{X}}\right)=\mathbf{h}\left(\mathbf{X}\right)+\mathbf{v}-\mathbf{h}\left(\hat{\mathbf{X}}\right)\simeq\mathbf{v}+\mathbf{H}\left(\hat{\mathbf{X}}\right)\left(\mathbf{X}^*-\hat{\mathbf{X}}\right). \tag{B.7}$$

Substituting equation (B.4) into (B.7) it is found that

$$\mathbf{r}=\mathcal{W}\mathbf{v}, \tag{B.8}$$

where

$$\mathcal{W} = \mathbf{I} - \mathbf{H}\left(\hat{\mathbf{X}}\right)\left[\mathbf{H}^T\left(\hat{\mathbf{X}}\right)\mathbf{H}\left(\hat{\mathbf{X}}\right)\right]^{-1}\mathbf{H}^T\left(\hat{\mathbf{X}}\right) \tag{B.9}$$

is the $M \times M$ residual sensitivity matrix. This sensitivity matrix is used mainly in gross error detection and identification procedures [24]. Under the Gaussian assumption for $\mathbf{v}$, the residual vector has zero mean and the covariance matrix is given by

$$\mathsf{E}\left[\mathbf{r}\mathbf{r}^T\right] = \mathcal{W}\mathbf{R}. \tag{B.10}$$

Based on Eq. (B.10) the covariance matrix for weighted residuals $\mathbf{r}_W$ is equal to $\mathcal{W}$, and for normalized residuals $\mathbf{r}_N$ is equal to $\mathbf{I}$.

Equation (B.8) can be partitioned as follows

$$\begin{pmatrix} \mathbf{r}_t \\ \mathbf{r}_s \end{pmatrix} = \begin{pmatrix} \mathcal{W}_{tt} & \mathcal{W}_{ts} \\ \mathcal{W}_{st} & \mathcal{W}_{ss} \end{pmatrix} \begin{pmatrix} \mathbf{v}_t \\ \mathbf{v}_s \end{pmatrix}, \tag{B.11}$$

where the subscript s refers to the maximum number $(M-n)$ of suspected data [24], and the subscript t indicates an $n \times 1$ vector containing the smallest residuals. Error $\mathbf{v}_s$ in the suspected measurements can be estimated by using LS from Eq. (B.11)

$$\hat{\mathbf{v}}_s = \mathcal{W}_{ss}^{-1}\mathbf{r}_s = \mathcal{W}_{ss}^{-1}\mathcal{W}_{st}\mathbf{v}_t + \mathbf{v}_s. \tag{B.12}$$

In some cases it is useful to find how $\mathcal{W}_{ss}^{-1}\mathcal{W}_{st}$ is related to $\mathbf{H}\left(\hat{\mathbf{X}}\right)$. First, let us make the partition of Jacobian matrix

$$\mathbf{H} = \begin{pmatrix} \mathbf{H}_t \\ \mathbf{H}_s \end{pmatrix}, \tag{B.13}$$

where s and t are defined above. We know that $\mathbf{H}$ is a function of $\hat{\mathbf{X}}$ in Eq. (B.13), but we are not showing that explicitly in order to simplify notation. Substituting Eq. (B.13) in Eq. (B.9) yields

$$\mathcal{W} = \mathbf{I} - \begin{pmatrix} \mathbf{H}_t \\ \mathbf{H}_s \end{pmatrix} \mathbf{G}^{-1} \begin{pmatrix} \mathbf{H}_t^T & \mathbf{H}_s^T \end{pmatrix} = \mathbf{I} - \begin{pmatrix} \mathbf{H}_t\mathbf{G}^{-1}\mathbf{H}_t^T & \mathbf{H}_t\mathbf{G}^{-1}\mathbf{H}_s^T \\ \mathbf{H}_s\mathbf{G}^{-1}\mathbf{H}_t^T & \mathbf{H}_s\mathbf{G}^{-1}\mathbf{H}_s^T \end{pmatrix}, \tag{B.14}$$

where

$$\mathbf{G} = \begin{pmatrix} \mathbf{H}_t^T & \mathbf{H}_s^T \end{pmatrix} \begin{pmatrix} \mathbf{H}_t \\ \mathbf{H}_s \end{pmatrix} = \mathbf{H}_t^T \mathbf{H}_t + \mathbf{H}_s^T \mathbf{H}_s. \tag{B.15}$$

According to equations (B.11) and (B.14) it is found that

$$\mathcal{W}_{ss}^{-1} \mathcal{W}_{st} = -\left(\mathbf{I} - \mathbf{H}_s \mathbf{G}^{-1} \mathbf{H}_s^T\right)^{-1} \mathbf{H}_s \mathbf{G}^{-1} \mathbf{H}_t^T. \tag{B.16}$$

Based on Sherman&Morrison matrix inversion formula we can find $\mathbf{G}^{-1}$ for $\mathbf{G}$ given in Eq. (B.15)

$$\mathbf{G}^{-1} = \left(\mathbf{H}_t^T \mathbf{H}_t\right)^{-1} - \left(\mathbf{H}_t^T \mathbf{H}_t\right)^{-1} \mathbf{H}_s^T \left[\mathbf{I} + \mathbf{H}_s \left(\mathbf{H}_t^T \mathbf{H}_t\right)^{-1} \mathbf{H}_s^T\right]^{-1} \mathbf{H}_s \left(\mathbf{H}_t^T \mathbf{H}_t\right)^{-1}. \tag{B.17}$$

Substituting Eq. (B.17) in Eq. (B.16), and multiplying with $\mathbf{H}_t \mathbf{H}_t^{-1}$ from the right side yields

$$\mathcal{W}_{ss}^{-1} \mathcal{W}_{st} = -\left[\mathbf{I} - \mathbf{A} + \mathbf{A} \left(\mathbf{I} + \mathbf{A}\right)^{-1} \mathbf{A}\right]^{-1} \left[\mathbf{I} - \mathbf{A} \left(\mathbf{I} + \mathbf{A}\right)^{-1}\right] \mathbf{H}_s \mathbf{H}_t^{-1}, \tag{B.18}$$

where

$$\mathbf{A} = \mathbf{H}_s \left(\mathbf{H}_t^T \mathbf{H}_t\right)^{-1} \mathbf{H}_s^T.$$

Then, using Neumann Lemma

$$\left(\mathbf{I} - \mathbf{A}\right)^{-1} = \lim_{k \to \infty} \sum_{i=0}^{k} \mathbf{A}^i$$

in Eq. (B.18) it is found that

$$\left[\mathbf{I} - \mathbf{A} + \mathbf{A} \left(\mathbf{I} + \mathbf{A}\right)^{-1} \mathbf{A}\right]^{-1} \left[\mathbf{I} - \mathbf{A} \left(\mathbf{I} + \mathbf{A}\right)^{-1}\right] = \mathbf{I},$$

and

$$\mathcal{W}_{ss}^{-1} \mathcal{W}_{st} = -\mathbf{H}_s \mathbf{H}_t^{-1}. \tag{B.19}$$

The concept of state estimation sensitivity to measurement errors (B.4) can be expanded to cover sensitivities of weighted estimation error of each measurement

$\varepsilon_i = \left(Z_{Ti} - \hat{Z}_i\right)/\sigma_i$, where Z_{Ti} is the true value of the measured quantity i. This estimation error can be derived using Eqs. (2.1) and (B.8), as follows

$$\boldsymbol{\varepsilon} = \mathbf{h}(\mathbf{X}) - \hat{\mathbf{Z}} = \mathbf{Z} - \mathbf{v} - \hat{\mathbf{Z}} = \mathbf{r} - \mathbf{v}, \tag{B.20}$$

$$\boldsymbol{\varepsilon} = -\boldsymbol{\Gamma}\mathbf{v}, \tag{B.21}$$

where $\boldsymbol{\Gamma}$ is the $M \times M$ sensitivity matrix

$$\boldsymbol{\Gamma} = \mathbf{I} - \mathcal{W} = \mathbf{H}\left(\hat{\mathbf{X}}\right)\left[\mathbf{H}^T\left(\hat{\mathbf{X}}\right)\mathbf{H}\left(\hat{\mathbf{X}}\right)\right]^{-1}\mathbf{H}^T\left(\hat{\mathbf{X}}\right), \tag{B.22}$$

It has been shown that these sensitivity matrices ($\mathbf{P}$, $\mathbf{W}$, and $\boldsymbol{\Gamma}$) are constant, i.e. not dependent on actual load situations [16].

B.2 Tracking State Estimation

B.2.1 The ANN State Estimator with Steepest Descent Dynamic

The ANN State Estimator with steepest descent dynamic given by Eq. (2.40), is represented as the system of non-linear differential equations

$$\tau_x \Delta\dot{\mathbf{X}} = \mathbf{f}(c\mathbf{u}) \tag{B.23}$$

$$\mathbf{u} = \mathbf{W}\Delta\mathbf{X} + \mathbf{U}. \tag{B.24}$$

Let us assume that the power system is in a steady state. Then the corresponding state estimator given by Eq. (B.23) is also in steady state, i.e. $\mathbf{f}(c\mathbf{u}_0) = \mathbf{0}$, with the constant input $\mathbf{U} = \mathbf{0}$. At time $t = t_0$, the input values of the estimator start to change according to the white noise process associated with the measurement errors. Excited by white noise the state deviation vector $\Delta\mathbf{X}$ starts to change, so that for $t > t_0$ we can write

$$\mathbf{U}(t) = \delta\mathbf{U}(t), \tag{B.25}$$

$$\Delta\mathbf{X}(t) = \delta\mathbf{X}(t). \tag{B.26}$$

The activation function $\mathbf{f}$ (given by Eq. (2.39)) is differentiable so that we can linearize the system of differential equations (B.23)

$$\delta\dot{\mathbf{X}} = \mathbf{F}\delta\mathbf{X} + \mathbf{G}\delta\mathbf{U}, \tag{B.27}$$

where

$$\mathbf{F} = \frac{1}{\tau_x}\frac{\partial \mathbf{f}(c\mathbf{u})}{\partial \Delta\mathbf{X}}\bigg|_{\substack{\Delta\mathbf{X}=0\\ \mathbf{U}=0}} = \frac{1}{\tau_x}\frac{2c}{L_u - L_l}\left(L_l^2\mathbf{I} - \mathbf{f}(c\mathbf{u}_0)^2\right)\mathbf{W}, \tag{B.28}$$

$$\mathbf{G} = \frac{1}{\tau_x}\frac{\partial \mathbf{f}(c\mathbf{u})}{\partial \mathbf{U}}\bigg|_{\substack{\Delta\mathbf{X}=0\\ \mathbf{U}=0}} = \frac{1}{\tau_x}\frac{2c}{L_u - L_l}\left(L_l^2\mathbf{I} - \mathbf{f}(c\mathbf{u}_0)^2\right). \tag{B.29}$$

If the limits of the activation functions $\mathbf{f}$ are $L_u = -L_l = L$, the formulas for calculating the $\mathbf{F}$ and $\mathbf{G}$ matrices of the linearized system (B.27) are

$$\mathbf{F} = \frac{1}{\tau_x}cL\mathbf{W}, \tag{B.30}$$

$$\mathbf{G} = \frac{1}{\tau_x}cL\mathbf{I}. \tag{B.31}$$

The solution of the linearized system (B.27) is

$$\delta\mathbf{X}(t) = \Phi(t, t_0)\,\delta\mathbf{X}_0 + \int_{t_0}^{t}\Phi(t, \tau)\,\mathbf{G}\delta\mathbf{U}(\tau)\,d\tau. \tag{B.32}$$

The ANN SE calculates the output $\Delta\mathbf{X}$ for each input sample at discrete moments in time t_q, $(q = 1, 2, \ldots)$. The input stays constant over the measurement sampling interval $\Delta t = t_{q+1} - t_q$. It follows that the dynamic ANN SE system can be represented with difference equations

$$\delta\mathbf{X}_{q+1} = \Phi_q\delta\mathbf{X}_q + \Gamma_q\delta\mathbf{U}_q, \tag{B.33}$$

where

$$\Phi_q = \exp(\mathbf{F}\Delta t), \qquad \Delta t = t_{q+1} - t_q, \tag{B.34}$$

$$\Gamma_q\delta\mathbf{U}_q = \int_{t_q}^{t_{q+1}}\Phi(t_{q+1}, \tau)\,\mathbf{G}\delta\mathbf{U}(\tau)\,d\tau = \mathbf{F}^{-1}\left(\exp(\mathbf{F}\Delta t) - \mathbf{I}\right)\mathbf{G}\delta\mathbf{U}_q. \tag{B.35}$$

In order to find the sensitivity vector $\mathbf{S}$, by using the definition (2.41), we first need to derive the formula for variances of output errors. The variances are diagonal elements of the covariance matrix given by

$$\mathrm{cov}\left[\delta \mathrm{X}\right] = \mathrm{E}\left[\delta \mathrm{X}_q \delta \mathrm{X}_q^T\right], \tag{B.36}$$

where the expectation E is going over q. Before we derive the expression for the covariance matrix, the difference equation (B.33) has to be uncoupled by introducing the coordinate rotation $\delta \bar{\mathbf{X}} = \mathbf{Q}^T \delta \mathbf{X}$. The matrix $\mathbf{Q}$ is the eigenvector matrix of the matrix $\mathbf{F}$. Since $\mathbf{F} = \mathbf{F}^T$ and all entries of $\mathbf{F}$ are real numbers, the corresponding eigenvalues must be real. The eigenvalues of $\mathbf{F}$ are elements of the diagonal eigenvalue matrix $\mathbf{\Lambda}$. In this case, the eigenvector matrix $\mathbf{Q}$ can be made orthonormal, i.e. $\mathbf{Q}\mathbf{Q}^T = \mathbf{I}$. It follows that $\mathbf{F} = \mathbf{Q}\mathbf{\Lambda}\mathbf{Q}^{-1} = \mathbf{Q}\mathbf{\Lambda}\mathbf{Q}^T$. Using these results from linear algebra, after the coordinate rotation $\delta \bar{\mathbf{X}} = \mathbf{Q}^T \delta \mathbf{X}$, we obtain the following system of difference equations

$$\delta \bar{\mathbf{X}}_{q+1} = \exp\left(\mathbf{\Lambda}\Delta t\right) \delta \bar{\mathbf{X}}_q + \mathbf{\Lambda}^{-1} \left(\exp\left(\mathbf{\Lambda}\Delta t\right) - \mathbf{I}\right) \mathbf{Q}^T \mathbf{G} \delta \mathbf{U}_q. \tag{B.37}$$

Now we can find the product $\delta \bar{\mathbf{X}}_{q+1} \delta \bar{\mathbf{X}}_{q+1}^T$ using Eq. (B.37). Then the expectation of the product $\mathrm{E}\left[\delta \bar{\mathbf{X}}_q \delta \bar{\mathbf{X}}_q^T\right]$, using the fact that the estimation error $\delta \bar{\mathbf{X}}_q$ at t_q and the input perturbation $\delta \mathbf{U}_q$ are uncorrelated (a consequence of the fact that $\delta \mathbf{U}_q$ is a white sequence), is found to be

$$\mathrm{E}\left[\delta \bar{\mathbf{X}}_q \delta \bar{\mathbf{X}}_q^T\right] = \exp\left(2\mathbf{\Lambda}\Delta t\right) \mathrm{E}\left[\delta \bar{\mathbf{X}}_q \delta \bar{\mathbf{X}}_q^T\right] + \mathbf{\Lambda}^{-2} \left(\exp\left(\mathbf{\Lambda}\Delta t\right) - \mathbf{I}\right)^2 \mathbf{Q}^T \mathbf{G} \mathrm{E}\left[\delta \mathbf{U}_q \delta \mathbf{U}_q^T\right] \mathbf{G}^T \mathbf{Q} \tag{B.38}$$

From Eq. (B.38) we obtain the covariance of $\delta \bar{\mathbf{X}}_q$ in terms of the covariance of the input perturbation

$$\mathrm{cov}\left[\delta \bar{\mathbf{X}}\right] = \left(\mathbf{I} - \exp\left(2\mathbf{\Lambda}\Delta t\right)\right)^{-1} \mathbf{\Lambda}^{-2} \left(\exp\left(\mathbf{\Lambda}\Delta t\right) - \mathbf{I}\right)^2 \mathbf{Q}^T \mathbf{G}\, \mathrm{cov}\left[\delta \mathbf{U}_q\right] \mathbf{G}^T \mathbf{Q}. \tag{B.39}$$

To map the above result into the original coordinate system we use

$$\mathrm{cov}\left[\delta \mathrm{X}\right] = \mathbf{Q}\, \mathrm{cov}\left[\delta \bar{\mathbf{X}}\right] \mathbf{Q}^T,$$

and

$$\operatorname{cov}[\delta \mathbf{U}_q] = \mathbf{H}^T \mathbf{R} \mathbf{H},$$

where $\mathbf{R} = \mathsf{E}\left[\mathbf{v}\mathbf{v}^T\right]$ as defined in Eq. (2.2), and $\mathbf{H}$ is the Jacobian matrix given by Eq. (2.7). For this coordinate rotation we also have to use the expression

$$\mathbf{\Lambda}^n = \left(\mathbf{Q}^T \mathbf{F} \mathbf{Q}\right)^n = \mathbf{Q}^T \mathbf{F}^n \mathbf{Q}.$$

Then, for matrices $\mathbf{F}$ and $\mathbf{G}$ in Eq. (B.39) (mapped into original coordinate system) expressions (B.30) and (B.31) are used. The final formula for $\operatorname{cov}[\delta \mathbf{X}]$ becomes

$$\operatorname{cov}[\delta \mathbf{X}] = \left(\mathbf{I} - \exp\left(2\frac{1}{\tau_x} cL\Delta t \mathbf{W}\right)\right)^{-1} \left(\exp\left(\frac{1}{\tau_x} cL\Delta t \mathbf{W}\right) - \mathbf{I}\right)^2 \mathbf{W}^{-2} \mathbf{H}^T \mathbf{R} \mathbf{H}. \tag{B.40}$$

Now, if we assume $\mathbf{R} = \sigma^2 \mathbf{I}$, from the definition (2.41) the statistical sensitivity vector of ANN SE with the steepest descent dynamic on measurement errors modelled as a white noise process, is

$$\mathbf{S} = \sqrt{\operatorname{diag}\left[\left(\mathbf{I} - \exp\left(2\frac{1}{\tau_x} cL\Delta t \mathbf{W}\right)\right)^{-1} \left(\exp\left(\frac{1}{\tau_x} cL\Delta t \mathbf{W}\right) - \mathbf{I}\right)^2 \mathbf{W}^{-2} \mathbf{H}^T \mathbf{H}\right]}. \tag{B.41}$$

The sensitivity measure of ANN SE with steepest descent dynamic can be derived as an average of the elements of the sensitivity vector

$$\bar{S} = \frac{1}{N} \sum_{i=1}^{N} S_i \tag{B.42}$$

If we simplify (B.41) by introducing approximation $\exp(\mathbf{F}\Delta t) \approx \mathbf{I} + \mathbf{F}\Delta t$ for small Δt, and if we fix the weights $\mathbf{W} = -\mathbf{H}^T \mathbf{H}$, a very simple sensitivity measure appears

$$\bar{S} \approx \sqrt{\frac{\Delta t c L}{2\tau_x}} \tag{B.43}$$

B.2.2 The ANN State Estimator with Hopfield-style Dynamic

The mathematical model of the ANN SE with Hopfield-style dynamic given by Eq. (2.57) is the system of non-linear differential equations, represented in the

matrix form

$$\tau_x \Delta\dot{\mathbf{X}} = -\Delta\mathbf{X} + \mathbf{f}(c\mathbf{u}) \tag{B.44}$$

where $\mathbf{u}$ is given in (B.24). The linearization of the system (B.44) is done by using the same procedure as in the previous Section B.2.1. The system of linearized differential equations in the state-space form is

$$\delta\dot{\mathbf{X}} = \mathbf{F}\delta\mathbf{X} + \mathbf{G}\delta\mathbf{U}, \tag{B.45}$$

If the limits of the activation functions are $L_u = -L_l = L$, and the linearization is done around the point $\Delta\mathbf{X} = \mathbf{0}$ and $\mathbf{U} = \mathbf{0}$, the expressions for the matrices $\mathbf{F}$ and $\mathbf{G}$ of the system (B.45) are

$$\mathbf{F} = \frac{1}{\tau_x}(-\mathbf{I} + cL\mathbf{W}), \tag{B.46}$$

$$\mathbf{G} = \frac{1}{\tau_x} cL\mathbf{I}. \tag{B.47}$$

Note that $\mathbf{F}$ is a symmetrical matrix and we can apply the same procedure described in Section B.2.1 to find $\mathrm{cov}[\delta\mathbf{X}(t)]$. The expression for covariance is

$$\begin{aligned} \mathrm{cov}[\delta\mathbf{X}] &= \left(\mathbf{I} - \exp\left(2\frac{1}{\tau_x}\Delta t\,(cL\mathbf{W} - \mathbf{I})\right)\right)^{-1} \cdot \\ &\left(\exp\left(\frac{1}{\tau_x}\Delta t\,(cL\mathbf{W} - \mathbf{I})\right) - \mathbf{I}\right)^2 (cL\mathbf{W} - \mathbf{I})^{-2} (cL)^2 \mathbf{H}^T\mathbf{R}\mathbf{H}. \end{aligned} \tag{B.48}$$

The sensitivity vector and the sensitivity measure are found from $\mathrm{cov}[\delta\mathbf{X}(t)]$ by using the definitions (2.41) and (B.42), respectively. Introducing the approximation $\exp(\mathbf{F}\Delta t) \approx \mathbf{I} + \mathbf{F}\Delta t$ for small Δt,we obtain the simplified formula for the sensitivity vector

$$\mathbf{S} \approx \sqrt{\frac{\Delta t\,(cL)^2}{2\tau_x}\,\mathrm{diag}\left[(\mathbf{I} - cL\mathbf{W})^{-1}\mathbf{H}^T\mathbf{H}\right]}. \tag{B.49}$$

Appendix C

Real Time Recurrent Learning

In this appendix the steps required to derive the learning algorithm for ANN State Estimator are presented. The learning will be accomplished by minimizing an error function E. This function is represented as

$$E = \frac{1}{2}\sum_{i=1}^{m} p_i r_i^2, \tag{C.1}$$

where r_i denotes measurement residuals

$$r_i = z_i - h_i\left(\hat{\mathbf{X}}\right), \quad i = 1 \ldots m, \tag{C.2}$$

and p represents the additional weighting factors.

A dynamical way of minimizing E is to let the system evolve in the space of weights $\mathbf{W}$ along a learning trajectory which descends against the gradient of E. Thus the dynamical equation for the element W_{rs} of the weight matrix $\mathbf{W}$ is

$$\tau_w \frac{dW_{rs}}{dt} = -\frac{\partial E}{\partial W_{rs}}, \tag{C.3}$$

where τ_w is a constant which defines the time-scale over which weight matrix $\mathbf{W}$ changes. Performing differentiations ($\partial E/\partial W_{rs}$ using Eq. (C.1) and $\partial r/\partial W_{rs}$ using Eq. (C.2)) in Eq. (C.3) we obtain the following equation:

$$\tau_w \frac{dW_{rs}}{dt} = -\sum_{i=1}^{m} p_i r_i \frac{\partial r_i}{\partial W_{rs}} = \sum_{i=1}^{m} p_i r_i \sum_{j=1}^{n} \frac{\partial h_i(\mathbf{X})}{\partial \hat{x}_j} \frac{\partial \hat{x}_j}{\partial W_{rs}}. \tag{C.4}$$

C.1 The ANN State Estimator with Steepest Descent Dynamic

The derivative of $\hat{x}_i$ with respect to W_{rs} in Eq. (C.4) is obtained by first noting that $\Delta\hat{x}_i$ is a fixed point of the ANN SE given by Eq.(2.40), i.e. differential equation

$$\frac{d\Delta\hat{x}_i}{dt} = f_i(\sum_{j=1}^{n} W_{ij}\Delta\hat{x}_j + U_i) = 0. \tag{C.5}$$

Function f_i is a hyperbolic tangent function. This function has zero $f(0) = 0$; it follows

$$\sum_{j=1}^{n} W_{ij}\Delta\hat{x}_j + U_i = 0. \tag{C.6}$$

The result of differentiating Eq. (C.6) with respect to W_{rs} is

$$\sum_{j=1}^{n}\left(\frac{\partial W_{ij}}{\partial W_{rs}}\Delta\hat{x}_j + W_{ij}\frac{\partial\Delta\hat{x}_j}{\partial W_{rs}}\right) = 0. \tag{C.7}$$

Now, since the elements of matrix $\mathbf{W}$ are independent, it follows that the partial derivative $\partial W_{ij}/\partial W_{rs}$ is equal to 1 if $i = r$ and $j = s$, and zero otherwise, i.e.

$$\frac{\partial W_{ij}}{\partial W_{rs}} = \delta_{ir}\delta_{js}, \tag{C.8}$$

where δ_{ij} is the Kronecker delta. We can simplify Eq. (C.7) by substituting Eq. (C.8) into Eq. (C.7) and performing the summation over j. The result is

$$\sum_{j=1}^{n} W_{ij}\frac{\partial\Delta\hat{x}_j}{\partial W_{rs}} = -\delta_{ir}\Delta\hat{x}_s. \tag{C.9}$$

If we write Eq. (C.9) for $i = 1 \ldots n$ in the matrix form we obtain

$$\begin{pmatrix} W_{11} & \cdots & W_{1r} & \cdots & W_{1n} \\ \vdots & \ddots & & & \\ W_{r1} & \cdots & W_{rr} & & \\ \vdots & & & & \\ W_{n1} & \cdots & & & W_{nn} \end{pmatrix} \begin{pmatrix} \frac{\partial\Delta\hat{x}_1}{\partial W_{rs}} \\ \vdots \\ \frac{\partial\Delta\hat{x}_r}{\partial W_{rs}} \\ \vdots \\ \frac{\partial\Delta\hat{x}_n}{\partial W_{rs}} \end{pmatrix} = - \begin{pmatrix} \delta_{1r}\Delta\hat{x}_s \\ \vdots \\ \delta_{rr}\Delta\hat{x}_s \\ \vdots \\ \delta_{nr}\Delta\hat{x}_s \end{pmatrix} = - \begin{pmatrix} 0 \\ \vdots \\ \Delta\hat{x}_s \\ \vdots \\ 0 \end{pmatrix}. \tag{C.10}$$

The result of multiplying both sides of Eq. (C.10) with the inverse of weight matrix $\mathbf{W}$ is

$$\frac{\partial \hat{x}_i}{\partial W_{rs}} = \frac{\partial \Delta \hat{x}_i}{\partial W_{rs}} = -\left(\mathbf{W}^{-1}\right)_{ir} \Delta \hat{x}_s. \tag{C.11}$$

Substituting Eq. (C.11) into Eq. (C.4) and deleting "hat", we obtain the following form

$$\tau_w \frac{dW_{rs}}{dt} = y_r \Delta x_s, \tag{C.12}$$

where y_r is defined by

$$y_r = -\sum_{i=1}^{m} p_i r_i \sum_{j=1}^{n} \frac{\partial h_i(\mathbf{X})}{\partial x_j} \left(\mathbf{W}^{-1}\right)_{jr}. \tag{C.13}$$

Equations (C.12) and (C.13) specify a formal learning rule for modifying the weights $\mathbf{W}$. Equation (C.13) requires a matrix inversion, and therefore this learning algorithm is not suitable for parallel implementation as an ANN. A parallel method for calculation y_r can be obtained by the introduction of an associated ANN dynamical system. To obtain this dynamical system first write (C.13) in the matrix form, for $r = 1 \ldots n$,

$$\begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix} = -\begin{pmatrix} W_{11} & \cdots & W_{n1} \\ \vdots & \ddots & \\ W_{1n} & \cdots & W_{nn} \end{pmatrix}^{-1} \begin{pmatrix} \sum_{i=1}^{m} p_i r_i \partial h_i / \partial x_1 \\ \vdots \\ \sum_{i=1}^{m} p_i r_i \partial h_i / \partial x_n \end{pmatrix}. \tag{C.14}$$

After multiplying both sides of Eq. (C.14) with $\mathbf{W}^T$ we obtain

$$0 = \sum_{j=1}^{n} W_{jr} y_r + \sum_{i=1}^{m} p_i r_i \frac{\partial h_i}{\partial x_r}. \tag{C.15}$$

Finally, we can make the observation that the solution of this linear equation is the fixed points of the dynamical system given by

$$\frac{1}{\tau_y} \frac{dy_r}{dt} = \left(\sum_{j=1}^{n} W_{jr} y_r + T_r \right), \tag{C.16}$$

where T_r is input defined by

$$T_r = \sum_{i=1}^{m} p_i r_i \frac{\partial h_i}{\partial x_r}. \tag{C.17}$$

C.2 The ANN State Estimator with Hopfield-style Dynamic

The fixed point $\Delta\hat{\mathbf{X}}$ of CTRNN given by Eq. (2.57) must satisfy the nonlinear algebraic equation

$$\Delta\hat{x}_i = f_i\left(\sum_{i=1}^{n} W_{ij}\Delta\hat{x}_j + U_i\right), \qquad i = 1\ldots n. \tag{C.18}$$

The result of differentiating Eq. (C.18) with respect to W_{rs} on both sides is

$$\frac{\partial\Delta\hat{x}_i}{\partial W_{rs}} = f_i'(u_i)\sum_{j=1}^{n}\left(\frac{\partial W_{ij}}{\partial W_{rs}}\Delta\hat{x}_j + W_{ij}\frac{\partial\Delta\hat{x}_j}{\partial W_{rs}}\right), \tag{C.19}$$

where $u_i = \sum_{j=1}^{n} W_{ij}\Delta\hat{x}_j + U_i$. We can simplify Eq. (C.19) by substituting Eq. (C.8) into it,and performing the summation over j on the right side of the equation. Also the left side can be expressed as the product of the identity matrix times the matrix of partial derivatives of $\Delta\hat{x}_j$. The result is

$$\sum_{j=1}^{n}\delta_{ij}\frac{\partial\Delta\hat{x}_j}{\partial W_{rs}} = f_i'(u_i)\left(\delta_{ir}\Delta x_s + \sum_{j=1}^{n} W_{ij}\frac{\partial\Delta\hat{x}_j}{\partial W_{rs}}\right), \tag{C.20}$$

where δ_{ij} is the Kronecker delta. After collecting all the partial derivatives in (C.20) on the left side, we obtain

$$\sum_{j=1}^{n} L_{ij}\frac{\partial\Delta\hat{x}_j}{\partial W_{rs}} = \delta_{ir}f_i'(u_i)\,\Delta\hat{x}_s \tag{C.21}$$

where $L_{ij} = \delta_{ij} - f_i'(u_i)\,W_{ij}$. Equation (C.21) is similar to Eq. (C.9), and we can write all equations for $i = 1\ldots n$ in the a matrix form

$$\begin{pmatrix} L_{11} & \cdots & L_{1r} & \cdots & L_{1n} \\ \vdots & \ddots & & & \\ L_{r1} & \cdots & L_{rr} & & \\ \vdots & & & & \\ L_{n1} & \cdots & & & L_{nn} \end{pmatrix}\begin{pmatrix} \frac{\partial\Delta\hat{x}_1}{\partial W_{rs}} \\ \vdots \\ \frac{\partial\Delta\hat{x}_r}{\partial W_{rs}} \\ \vdots \\ \frac{\partial\Delta\hat{x}_n}{\partial W_{rs}} \end{pmatrix} = \begin{pmatrix} \delta_{1r}f_1'(u_1)\,\Delta\hat{x}_s \\ \vdots \\ \delta_{rr}f_r'(u_r)\,\Delta\hat{x}_s \\ \vdots \\ \delta_{nr}f_n'(u_n)\,\Delta\hat{x}_s \end{pmatrix} = \begin{pmatrix} 0 \\ \vdots \\ f_r'(u_r)\,\Delta\hat{x}_s \\ \vdots \\ 0 \end{pmatrix}. \tag{C.22}$$

If we multiply both sides of Eq. (C.22) by $\mathbf{L}^{-1}$ the result is

$$\frac{\partial \hat{x}_i}{\partial W_{rs}} = \frac{\partial \Delta \hat{x}_i}{\partial W_{rs}} = \left(\mathbf{L}^{-1}\right)_{ir} f_r'\left(u_r\right) \Delta \hat{x}_s \tag{C.23}$$

Substituting Eq. (C.23) into Eq. (C.4) and deleting "hat", we obtain the dynamic equation

$$\tau_w \frac{dW_{rs}}{dt} = y_r \Delta x_s, \tag{C.24}$$

where y_r is defined by

$$y_r = f_r'\left(u_r\right) \sum_{i=1}^{m} p_i r_i \sum_{j=1}^{n} \frac{\partial h_i\left(\mathbf{X}\right)}{\partial x_j} \left(\mathbf{L}^{-1}\right)_{jr.} \tag{C.25}$$

Finally, using the same approach as in Section C.1, we develop an ANN dynamical system to solve Eq. (C.25) without matrix inversion

$$\tau_y \frac{dy_r}{dt} = -y_r + f_r'\left(u_r\right) \left(\sum_{j=1}^{n} W_{jr} y_j + T_r \right), \qquad r = 1 \ldots n, \tag{C.26}$$

where T_r is the input defined by Eq. (C.17).

Appendix D

Data for Test Power Systems

D.1 3-Bus Power System

The 3-bus test system with disposition of meters is shown in Figure D.1. All measurements are active power measurements; #1, #2 and #3 are active power flow and #4 is active power injection. We assume that all voltages are equal to $1pu$. Susceptances of the lines are equal to: $B_{12} = B_{23} = 1pu$ and $B_{31} = 1.5pu$.The observation equations are linear and given here in the matrix form

$$\mathbf{Z} = \mathbf{HX}$$

where

$$\mathbf{H} = \begin{pmatrix} 1 & -1.5 & 0 & -1.5 & 2.5 \\ -1 & 0 & -1 & -1 & -1 \end{pmatrix}^T.$$

The last row in $\mathbf{H}$ corresponds to zero injection measurement (virtual measurement) at node 1. There are two states comprising phase angles at nodes 1 (x_1) and 2 (x_2). The phase angle at node 3 is considered as a reference. Initial values of the states and corresponding measurements are given in Table D.1.

x_1	x_2	z_1	z_2	z_3	z_4
-0.25	-0.625	0.375	0.375	0.625	1

Table D.1: *Initial data for 3-bus test system*

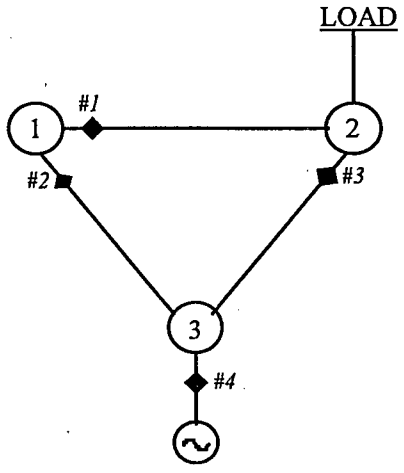

Figure D.1: *3-bus test system*

Node No.	2	3	4	5	6	9	10	11	12	13	14
MW	18.66	81.01	41.11	6.54	9.63	25.37	7.32	2.85	4.68	10.35	11.42
MVAr	10.92	16.34	-3.35	1.38	6.45	12.28	4.71	1.47	1.23	4.44	3.83

Table D.2: *Initial load values for 14-Bus Network*

D.2 IEEE14-Bus Power System

Figure D.2 shows the manner in which measurements are disposed in the 14-bus power system [86]. The measurement set is composed of 67 observations including: active and reactive power flows ($\bullet$), active and reactive power injections ($\triangleright$), and voltage magnitudes ($\times$). Voltage magnitude measurements are assumed to have errors normally distributed with the standard deviation equal to $0.007pu$. Normally distributed errors of power flow and power injection measurements have the standard deviation equal to $0.02pu$. Initial load values are given in Table D.2, and power system parameters in Table D.3.

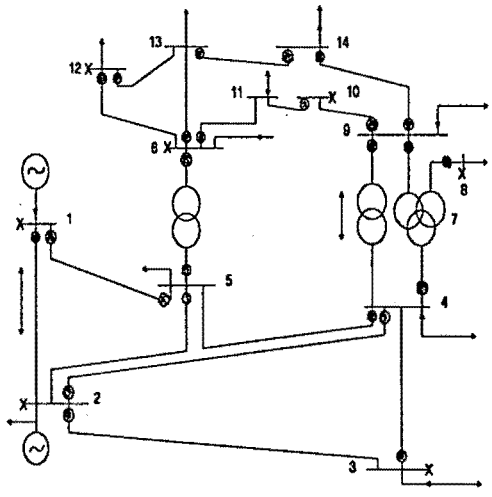

Figure D.2: *14-Bus Network*

Line Designation	Resistance [p.u.]	Reactance [p.u.]
1-2	0.01938	0.05917
1-5	0.05403	0.22304
2-3	0.04699	0.19797
2-4	0.05811	0.17632
2-5	0.05695	0.17388
3-4	0.06701	0.17103
4-5	0.01335	0.04211
4-7	0	0.20912
4-9	0	0.55618
5-6	0	0.25202
6-11	0.09498	0.1989
6-12	0.12291	0.25581
6-13	0.06615	0.13027
7-8	0	0.17615
7-9	0	0.11001
9-10	0.03181	0.08450
9-14	0.12711	0.27038
10-11	0.08205	0.19207
12-13	0.22092	0.19988
13-14	0.17093	0.34802

Table D.3: *Data for the IEEE 14-Bus Network (p.u. on 100MVA base)*

Bibliography

[1] L.H.Fink and K.Carlsen, "Operating under stress and strain", IEEE Spectrum, March 1978, pp.48-53.

[2] T.E. Dy Liacco, "System security: the computer's role", IEEE Spectrum, June 1978, pp.43-50.

[3] A.S.Debs and A.R.Benson, "Security assessment of power systems", Proceedings of Engineering Foundation Conference on System Engineering for Power, Henniker, August 1975, pp.144-176.

[4] B.Stott, O.Alsac and A.Monticelli, "Security analysis and optimization", Proceedings of the IEEE, Vol. 75, No. 12, December 1987, pp.1623-1644.

[5] F.F.Wu, "Real-time network security monitoring, assessment and optimization", International Journal of Electric Power and Energy Systems, Vol. 10, No. 2, April 1988.

[6] F.Schweppe, J.Wildes and D.Rom, "Power system static state estimation, Parts I, II, III", IEEE Transactions on Power Apparatus and Systems, Vol. PAS-89, January 1970, pp.120-135.

[7] E.Handschin and A.Petroianu, "Energy Management Systems", Springer-Verlag, 1991.

[8] A.G.Phadke, J.S.Thorp and K.J.Karimi, "State estimation with phasor measurements", IEEE Transactions on Power Systems, Vol. PWRS-1, February 1986, pp.233-241.

[9] F.C.Aschmoneit, N.M.Peterson and E.C.Adrian, "State estimation with quality constraints", Proceedings of 10th PICA Conference, May 1977, pp.427-430.

[10] A.Monticelli and A.Garcia, "Fast decoupled state estimators", IEEE Transactions on Power Systems, Vol. 5, No. 2, May 1990, pp.556-564.

[11] R.G.Wasley and W.I.Stadlin, "Network applications in energy management system", IEEE Computer Applications in Power, January 1991, pp.31-36.

[12] T.E. Dy Liacco, "The role and implementation of state estimation in an energy management system", International Journal on Electric Power and Energy Systems, Vol. 12, No. 2, April 1990, pp.75-79.

[13] G.R.Krumholz, K.A.Clements and P.W.Davis, "Power system observability: a practical algorithm using network topology", IEEE Transactions on Power Apparatus and Systems, Vol. PAS-99, July/August 1980, pp.1534-1542.

[14] A.Monticelli and F.F.Wu, "Network observability: theory", IEEE Transactions on Power Apparatus and Systems, Vol. PAS-104, May 1985, pp.1042-1048.

[15] A.Bose and K.A.Clements, "Real-time modelling of power network", Proceedings of the IEEE, Vol. 75, No. 12, December 1987, pp.1606-1622.

[16] H.J.Koglin at al., "Local redundancy and identification of bad data", Proceedings of the 9th Power Systems Computation Conference, Cascais 1987, pp.528-534.

[17] L.Mili, Th. Van Cutsem and M.Pavella, "Bad data identification methods in power system state estimation - a comparison study", IEEE Transactions on Power Apparatus and Systems, Vol. PAS-104, November 1985, pp.3037-3049.

[18] R.D.Masiello and F.C.Schweppe, "A tracking static state estimator", IEEE Transactions on Power Apparatus and Systems, Vol. PAS-90, March/April 1971, pp.1025-1033.

[19] A.M. Leite da Silva, M.B. Do Coutto Filho and J.F. de Queiroz, "State forecasting in electric power systems", IEE Proceedings C, Vol. 130, September 1983, pp.237-244.

[20] "Parallel Multiarea State Estimation", EPRI Research Project EL-2218, January 1982.

[21] Nakagawa, at al., "Neural network application to state estimation computation", First International Forum on Applications of Neural Networks to Power Systems, Seatle WA, July 1991, pp.188-194.

[22] R.Zivanovic and A.Petroianu, "Application of neural computing to the state estimation problem", IEEE International Conference on Power System Technology, October 1994, Beijing, China.

[23] A.Monticelli, F.F.Wu and M.Yen, "Multiple bad data identification for state estimation by combinatorial optimization", IEEE Transactions on Power Delivery, Vol. PWRD-1, No. 3, 1986, pp.361-369.

[24] L.Mili, Th. Van Cutsem and M.Pavella, "Hypothesis testing identification: a new method for bad data analysis in power system state estimation", IEEE Transactions on Power Apparatus and Systems, Vol. PAS-103, No. 11, 1984, pp.3239-3252.

[25] H.M.Merill and F.C.Schweppe, "Bad data suppression in power system static state estimation", IEEE Transactions on Power Apparatus and Systems, Vol. PAS-90, No. 6, 1971, pp.2718-2725.

[26] E.Handschin, F.C.Schweppe, J.Kohlas and A.Fiechter, "Bad data analysis for power system state estimation", IEEE Transactions on Power Apparatus and Systems, Vol. PAS-94, No. 2, 1975, pp.329-337.

[27] M.R.Irving, R.C.Owen and M.Sterling, "Power system state estimation using linear programming", Proceedings of IEE, Vol.125, No. 9, September 1978, pp.879-885.

[28] W.W.Kotiuga, "Development of a least absolute value power system tracking state estimator", IEEE Transactions on Power Apparatus and Systems, Vol. PAS-104, No. 5, May 1985, pp.1160-1166.

[29] D.M.Falcao and S.M. de Assis, "Linear programming state estimation: error analysis and gross error identification", IEEE Transactions on Power Systems, Vol. 3, No. 3, August 1988, pp.809-815.

[30] A.Abur, "A bad data identification method for linear programming state estimation", IEEE Transactions on Power Systems, Vol. 5, No. 3, August 1990, pp.894-901.

[31] M.K.Celik and A.Abur, "Use of scaling in WLAV estimation of power system states", IEEE Transactions on Power Systems, Vol. 7, No. 2, May 1992, pp.684-692.

[32] A.A. El Keib and H.Singh, "Fast linear programming state estimation using the dual formulation", IEEE Transactions on Power Systems, Vol. 7, No.2, May 1992, pp.620-628.

[33] L.Mili, V.Phaniraj and P.J.Rousseeuw, "Robust estimation theory for bad data diagnostics in electric power systems", Control and Dynamic Systems: Advances in Theory and Applications, Vol.37: Advances in Industrial Systems, edited by C.T.Leondes, Academic Press, 1990.

[34] R.Zivanovic and A.Petroianu, "An algorithm for robust estimation in electric power systems", IFAC Control of Power Plants and Power Systems, SIPOWER'95, Cancun, Mexico 1995, pp.89-94.

[35] R.Zivanovic and A.Petroianu, "An algorithm for bad data processing in a linear programming based state estimation", IEEE Power Tech Conference, paper No. SPT IC 05-02-0600, Stocholm 1995.

[36] L.Mili, V.Phaniraj and P.J.Rousseeuw, "Least median of squares estimation in power systems", IEEE Transactions on Power Systems, Vol. 6, No.2, May 1991, pp.511-523.

[37] L.Mili at al., "Robust state estimation based on projection statistics", IEEE/PES Winter Meeting, paper No. 95 WM 216-2 PWRS, January 1995.

[38] M.M.Adibi, at al., "Error reduction in state estimator measurements and modeling parameters", Proceedings of 11th PSCC, Avignon, France 1993.

[39] A.S.Debs, "Estimation of steady-state power system model parameters", IEEE Transactions on Power Apparatus and Systems, Vol. PAS-93, 1974, pp.1260-1268.

[40] Th. Van Cutsem and V.H.Quintana, "Network parameter estimation using online data with application to transformer tap position estimation", IEE Proceedings C, Vol. 135, No. 1, January 1988, pp.31-40.

[41] A.S.Debs and R.E.Larson, "A dynamic estimator for tracking the state of a power system", IEEE Transactions on Power Apparatus and Systems, Vol. PAS-89, No. 7, 1970, pp.1670-1678.

[42] P.Rousseaux, Th. Van Cutsem and T.E. Dy Liacco, "Whether dynamic state estimation?", International Journal on Electric Power and Energy Systems, Vol. 12, No. 2 April 1990, pp.104-116.

[43] A.M. Lite da Silva, M.B. do Coutto Filho and J.M.C.Cantera, "An efficient dynamic state estimation algorithm including bad data processing", IEEE Transactions on Power Systems, Vol. PWRS-2, No. 4, 1987, pp.1050-1058.

[44] R.Lippmann, "An introduction to computing with neural nets", IEEE ASSP Magazine, April 1987, pp.4-22.

[45] D.R.Hush and B.G.Horne, "Progress in supervised neural networks", IEEE Signal Processing Magazine, January 1993, pp.8-39.

[46] P.J.Werbos, "Beyond regression: new tools for prediction and analysis in the behavioral sciences", Doctoral Dissertation, Applied Mathematics, Harvard University, Boston MA, November 1974.

[47] D.E.Rumelhart and J.L.McClelland, "Parallel Distributed Processing: Explorations in the Microstructure of Cognition, MIT Press, Cambrige MA, 1986.

[48] J.J.Hopfield, "Neural networks and physical systems with emergent collective computational abilities," Proceedings of the National Academy of Science USA, 1982, pp.2554-2558.

[49] F.J.Pineda, "Dynamics and architecture for neural computation", Journal of Complexity, 1988, pp.216-245.

[50] R.J.Williams and D.Zipser, "A learning algorithm for continually running fully recurrent neural networks",Neural Computation, 1989, pp.270-280.

[51] C.Mead, "Analog VLSI and Neural Systems", Addison Wesley, New York, 1989.

[52] M.P.Kennedy and L.O.Chua, "Neural networks for non-linear programing", IEEE Transactions on Circuits and Systems, Vol. 35, No. 5, May 1988, pp.554-562.

[53] A.R.Vasquez, at al., "Non-linear switched-capacitor "neural" networks for optimization problems", IEEE Transactions on Circuits and Systems, Vol. 37, No. 3, March 1990, pp.384-397.

[54] R.D.Castro, at al., "Analog neural programmable optimizers in CMOS VLSI technologies", IEEE Journal of Solid-State Circuits, Vol. 27, No. 7, July 1992, pp.1110-1115.

[55] 80170NX Electronically Trainable Analog Neural Network, Intel Corp., 1993.

[56] T.S.Dillon, "Artificial neural network applications to power systems and their relationship to symbolic methods", International Journal on Electric Power and Energy Systems, Vol. 13, No. 2, April 1991, pp.66-72.

[57] D.J.Sobajic and Y.H.Pao, "Artificial neural-net based dynamic security assessment for electric power systems", IEEE Transactions on Power Systems, Vol. 4, No. 4, February 1989, pp.220-228.

[58] V.Sagar, at al., "Artificial neural networks and their applications to power systems - a bibliographical survey", Electric Power Systems Research, 28 (1993), pp.67-79.

[59] M.H.Sendaula, S.K.Biswas, A.Eltom, C.Parten and W.Kazibwe, "Simultaneous solution of unit commitment and dispatch problems using artificial neural networks", International Journal on Electric Power and Energy Systems, Vol. 15, No.3, June 1993, pp.193-200.

[60] S.Matsuda and Y.Akimoto, "The representation of large numbers in neural networks and its application to economical load dispatching of electric power", Proceedings of IJCNN-89, Washington DC, June 1989, pp.587-592.

[61] Y.Hayashi, at al., "Introduction of neural network theory to Newton-Raphson load flow", Proceeding of 3th Symposium on Expert Systems Application to Power Systems, April 1991, Tokyo-Kobe, Japan.

[62] H.Mori, N.Kitani and S.Tsuzuki, "Optimal power flow calculation using the Hopfield net", Proceeding of 3th Symposium on Expert Systems Application to Power Systems, April 1991, Tokyo-Kobe, Japan.

[63] T.T.Nguyen, "Neural network load flow", IEE Proceedings on Gener.Transm.Distrib., Vol.142, No. 1, January 1995.

[64] A.J.Kanekar and A.Feliachi, "State Estimation using Artificial Neural Networks", Proc. of 22nd Southeastern Symp. on System Theory, Los Alamitos 1990, CS Press, CA, pp. 552-556.

[65] A.P.Alves da Silva and V.H.Quintana, "Pattern analysis in power system state estimation", International Journal of Electric Power and Energy Systems, Vol. 17, No. 1, February 1995, pp.51-61.

[66] J. Ting-Ho Lo, "Synthetic Approach to Optimal Filtering", IEEE Transactions on Neural Networks, Vol. 5, No. 5, September 1994, pp.803-811.

[67] S. Osowski, "Neural Network for Non-Linear Programming with Linear Equality Constraints", International Journal of Circuit Theory and Applications", Vol. 20, 1992, pp.93-98.

[68] E.Mjolsness and C.Garrett, "Algebraic Transformations of Objective Functions", Neural Networks, Vol. 3, 1990, pp.651-669.

[69] P.E.Gill, W.Murray, and M.H.Wright, "Practical Optimization", Academic Press, 1981.

[70] M.M.Adibi at al., "Remote measurement calibration", IEEE Computer Applications in Power, October 1990, pp.37-42.

[71] A.Geleb, at al., "Applied Optimal Estimation", MIT Press, 1992.

[72] A.J.Wood and B.F.Wollenberg, Power Generation, Operation, and Control", Johan Wiley & Sons, 1984.

[73] F.F.Wu, "Power System State Estimation: a survey", Electric Power & Energy Systems, Vol. 12, No. 2, April 1990.

[74] J.M.Ortega and W.C.Rheinboldt, "Iterative Solution of Nonlinear Equations in Several Variables", Academic Press, 1970.

[75] B.Widrow and S.D.Stearns, "Adaptive Signal Processing" Prentice-Hall, 1985.

[76] J.Hertz, A.Krogh, and R.G.Palmer, "Introduction to the Theory of Neural Computation", Addison-Wesley, 1991.

[77] R.Zivanovic and A.Petroianu, "Neural Networks as a Power System State Estimation Tool", Southern African Universities Power Engineering Conference, 1994.

[78] D.M.Falcao, at al., "Power system tracking state estimation and bad data processing", IEEE Transactions on Power Apparatus and Systems, Vol. PAS-101, No.2, February 1982, pp. 325-333.

[79] A.Gjelsvik, at al., "Hachtel's Augmented Matrix Method - A Rapid Method Improving Numerical Stability in Power System Static State Estimation", IEEE Transactions on PAS, Vol. PAS-104, No. 11, November 1985, pp.2987-2993.

[80] Jin Young Choi and Chong-Ho Choi, "Sensitivity analysis of multilayer perceptron with differentiable activation function", IEEE Transactions on Neural Networks, Vol. 3, No. 1, January 1992, pp.101-107.

[81] S.Tam, at al., "Learning on an Analog VLSI Neural Network Chip", Proceedings of the 1990 IEEE International Conference on Systems, Man, and Cybernetics, November 1990, pp.701-703.

[82] A.G.Phadke at al., "A new measurement technique for tracking voltage phasors, local system frequency, and rate of change of frequency", IEEE Transactions on PAS, Vol. PAS-102, No. 5, May 1983, pp.1025-1038.

[83] C.Slivinsky at al., "Phase angle measurement applications: A new transducer that measures state variables in real time", 19th Annual Western Protective Relay Conference, Washington, October 1992.

[84] Ph.Denys at al., "Measurement of voltage phase angle for the French future defence plan against losses of synchronism", IEEE Transactions on Power Delivery, Vol. 7, No. 1, January 1992.

[85] J.S.Thorp, at al., "Real-time voltage phasor measurements for static state estimation", IEEE Transactions on Power Apparatus and Systems, Vol. PAS-104, No. 11, November 1985, pp.3098-3106.

[86] Y.Wallach, "Calculations and programs for power system networks", Prentice-Hall, Inc., 1986.