NOTICE

The quality of this microform is heavily dependent upon the quality of the original thesis submitted for microfilming. Every effort has been made to ensure the highest quality of reproduction possible.

If pages are missing, contact the university which granted the degree.

Some pages may have indistinct print especially if the original pages were typed with a poor typewriter ribbon or if the university sent us an inferior photocopy.

Reproduction in full or in part of this microform is governed by the Canadian Copyright Act, R.S.C. 1970, c. C-30, and subsequent amendments.

AVIS

La qualité de cette microforme dépend grandement de la qualité de la thèse soumise au microfilmage. Nous avons tout fait pour assurer une qualité supérieure de reproduction.

S'il manque des pages, veuillez communiquer avec l'université qui a conféré le grade.

La qualité d'impression de certaines pages peut laisser à désirer, surtout si les pages originales ont été dactylographiées à l'aide d'un ruban usé ou si l'université nous a fait parvenir une photocopie de qualité inférieure.

La reproduction, même partielle, de cette microforme est soumise à la Loi canadienne sur le droit d'auteur, SRC 1970, c. C-30, et ses amendements subséquents.

Canada

Fault Identification and Fault Location in Series Compensated Transmission Lines using Neural Networks

Tiberiu Grigoriu

A Thesis

in

The Department

of

Electrical and Computer Engineering

Presented in Partial Fulfilment of the Requirements

for the Degree of Master of Applied Science at

Concordia University

Montreal, Quebec, Canada

August 1993

ISBN 0-315-90928-5

Canada

ABSTRACT

Fault Identification and Fault Location in Series Compensated Transmission Lines using Neural Networks

Tiberiu Grigoriu

Changing requirements and increasing complexity of power systems demand new techniques to improve the reliability and protection of power systems. Traditional methods of transmission line protection based on distance relays are increasingly becoming strained since the occurrence of a fault in a power system produces three types of switching transients: at 60 Hz (the power frequency), at higher frequencies caused by resonance of the system's LC elements and very high frequency transients due to lightening surges. These transients have a magnitude and a rate of decay that depend on many factors such as fault location, fault type, and system parameters. These transients make the relationships between the bus voltages and line currents not easily definable, and many times the traditional protection schemes fail to operate properly.

Additional difficulties are encountered for series compensated transmission lines.

The increase of transmitted power is achieved by compensation of the line impedance by adding a series capacitance. The secondary effects of this series capacitance are

1. the creation of new switching transients due to the resonance between the capacitive impedance of the series capacitor and the inductive impedance of the line
2. an abrupt change in the linear impedance of the transmission line.

These secondary effects contribute to the distortion of the relationships between bus voltages and line currents, and confuses the traditional protection schemes. This topic is discussed in section 1.3.

Artificial Neural Networks (ANNs) might offer a novel solution to detect a fault in a power system since ANNs are able to extract the characteristic features from the bus voltages and line currents. There have been previous attempts to use ANNs in protective relaying schemes [7] or monitoring of power systems [10, 14]. Also some attempts have been made to combine ANN with Artificial Intelligence and Expert Systems [12, 13].

This thesis proposes a novel approach to the protection of series compensated transmission lines by using an ANN for the identification and location of a fault. Two different architectures for ANN (Counter Propagation Network (CPN), and

Adaptive Resonant Theory (ART)) are considered and a comparison of their performances is made. The bus voltages and line currents are pre-processed using different normalization methods. ANNs are then employed to classify the feature-vector space into distinct categories.

Dedication

To my parents Mr. Constantin Grigoriu, Mrs. Silvia Grigoriu and my brother Mr. Octavian Grigoriu with love.

ACKNOWLEDGEMENTS

I take this opportunity to thank all my supervisors and friends, Dr. Rajnikant Patel, Dr. Vijay Sood and Dr. Khashayar Khorasani. I thank them for their financial support, guidance and patience in preparing me for this thesis.

I also wish to thank all the members of the Computer support group. Nicky Ayoub, Gustavo Vegas, Dave Kai-Chui Chu and Guy Gosselin have been extremely helpful and very responsive in solving my endless problems with the computer resources.

I would also like to thank all my friends and fellow graduate students for the mental support, particularly Nahi Kandil, Vijay Khatri, Jerzy Bloch and Zhengcheng Lin.

Contents

List of Figures

List of Tables

Chapter 1

Protection of Power Systems

1.1 Introduction

A power system consists of a network of generators that send power to consumers through transmission lines. Due to natural or other causes, i.e. lightening, or malfunctions of some components, fault conditions can occur anywhere in the power system. Such fault conditions can result in severe damage to the system equipment due to either overvoltage and/or overcurrent stress. Thus, all the components of the power system need to be protected. The function of the protection system is to isolate the faulted component from the remaining power system, quickly and in a safe manner, and restore power transmission.

As the complexity of power systems has increased proportionally to the number of consumers and power demand, faster and more efficient methods for monitoring faults are needed. Fault monitoring techniques are required to rapidly and accurately identify the fault type and the fault location. An accurate fault identification ensures a correct protection sequence to restore the system stability. Similarly, identification of an exact fault location ensures the rapid isolation and restoration of the system.

In traditional protection methods, the fault monitoring task has been accomplished by estimating the fundamental frequency components from the corrupted voltage and current signals following the fault occurrence. By this analyses, the faulted phases may be determined. The following types of faults are relatively easy

to identify: Single Line to Ground (SLG) and Three Line to Ground (3LG) faults. However, there are usually difficulties in distinguishing between high impedance faults such as Double Line to Ground (DLG) and Double Line (DL) faults.

In general, the protection schemes use as input data the phase voltage and current at the location of the protection device. Also, there are instances where information from other protection devices may be available using telecommunications. In this case, more reliable decisions can be taken, by comparing the information. However, the use of telecommunication channels to exchange information increases the cost and speed of operation. Furthermore, consideration of the lack of telecommunication should not compromise the protection scheme.

Different protection schemes, which rely on the information at either one or both ends of the transmission line are discussed in section 1.2.

1.2 Protection Schemes

Protection schemes based on the distance relay are among the most well-known [1, 2, 3]. Conventional distance relays measure the voltage and current at a relay location. Based on these measurements, the relays can identify the faulted phases. An approximation for the fault location can be made by computing the impedance

of the line. However, despite a relatively robust and simple implementation there are several drawbacks of this method:

- relay operation is based on the measurement of power frequency voltage and current components and therefore a relative long time delay is required,
- measurement of impedance is only meaningful in relation to steady-state sinusoidal conditions, and it becomes increasingly difficult to perform accurate and discriminative measurement for faulted conditions,
- errors of measurement are encountered during high-resistance earth faults and power swings,
- low apparent frequency of aperiodic components in the measurands, associated with faults in long-line applications, introduce an error in measurements.

These drawbacks resulted in new proposals for protection schemes. As the capabilities of microprocessor-based instrumentation increased, new schemes were developed for fast fault identification and location by the analyses of the transient components set-up by faults. The schemes operate with correlation techniques to recognize transient components that depart from the relaying point and return to it later after a direct reflection from the fault. The location of the fault is found by estimating the time between the departure and arrival of these signals. In contrast with traditional methods, indication of fault location is achieved faster [6, 4].

Fiber optics can also be used to identify the location of the fault [15]. The protection system uses a composite fiber optic Overhead Protection Ground Wire (OPGW). OPGW methods have both the lightning protection scheme and the information transmission function included within them. The principle used by this technique is based on a reversal of the direction of the OPGW current at the fault point. Fault sensors along the line extract information required for fault location. If the phase angle between adjacent sensors differs by about 180 degrees, the fault might occur in the section between them. This method could only detect ground faults, and errors might occur due to malfunction of the sensors.

In modern power systems, the use of series compensated transmission lines is becoming common. Such lines permit the transmission of more power than uncompensated lines. These lines pose difficult protection problems due to the presence of the lumped capacitance in the transmission line and usually different protection techniques are used. These are summarized in the next section.

1.3 Protection of series compensated transmission lines

Compensation is achieved by inserting a bank of capacitors in series with the transmission line. Typically, the capacitors are placed either in the middle or at the ends

of the line. The capacitive reactance of the series capacitors compensates for the inductive reactance of the line. This reduces the voltage drop caused by the latter. Therefore, long transmission lines become electrically shorter and the power transmission can be raised closer to the thermal limits of the line conductor.

Consider the feeder circuit in Fig. 1.1 with its the corresponding voltage-phasor diagram. The expression of the voltage drop V_D along the feeder has the following form,

$$V_D = IRcos\phi + IX_Lsin\phi \tag{1.1}$$

where R - resistance of the feeder circuit

X_L - inductive reactance of the feeder circuit

ϕ - power-factor angle

Since the resistance R is small, the magnitude of the second term in eq. 1.1 is greater than the first term. The difference between the two increases as the power factor becomes smaller and the ratio R/X_L becomes smaller.

Inserting a capacitor C in series with the line (Fig. 1.1) alters equation 1.1 to

$$V_D = IRcos\phi + I(X_L - X_C)sin\phi \tag{1.2}$$

where X_C - capacitive reactance of the series capacitor

From eq. 1.2, the second term of the eq. 1.1 is decreased by X_C. Therefore, the voltage drop V_D decreases and more power can be transmitted without changing the

power factor ϕ. Moreover, it is possible to overcompensate the line by choosing a value of capacitive reactance X_C greater than the inductive reactance X_L.

The transferred power P is given by

$$P = 3R(\tilde{V}_S - \tilde{\Gamma}_R) = \frac{3|\tilde{V}_S||\tilde{V}_R|sin\phi}{(X_L - X_C)} \tag{1.3}$$

Fig. 1.2 shows a plot of equation 1.3 with increasing capacitive reactance compensation $X_{C_1} < X_{C_2} < X_{C_3}$. For $X_C = 0$, the available power P_{avail} cannot be transmitted across the line because X_L is too large. However, as the capacitive reactance is increased more power can be transmitted.

Inserting a capacitor in series with the transmission line, however, has side effects. The transmission line is a uniformly distributed impedance, and the presence of a lumped capacitor somewhere along it creates a distortion. The impedance of the line is no longer linear and therefore, difficulties in location of the fault using measurements of the impedance can be expected.

Furthermore, the resonance phenomena introduced in the transmission system by shunt-capacitors and shunt-reactors now has an additional component due to the series capacitor. This resonance involves exchange of energy between the series capacitance and the series inductance of the line and generators. This resonance frequency will increase from zero as the level of compensation increases, being highest when all the generation is on.

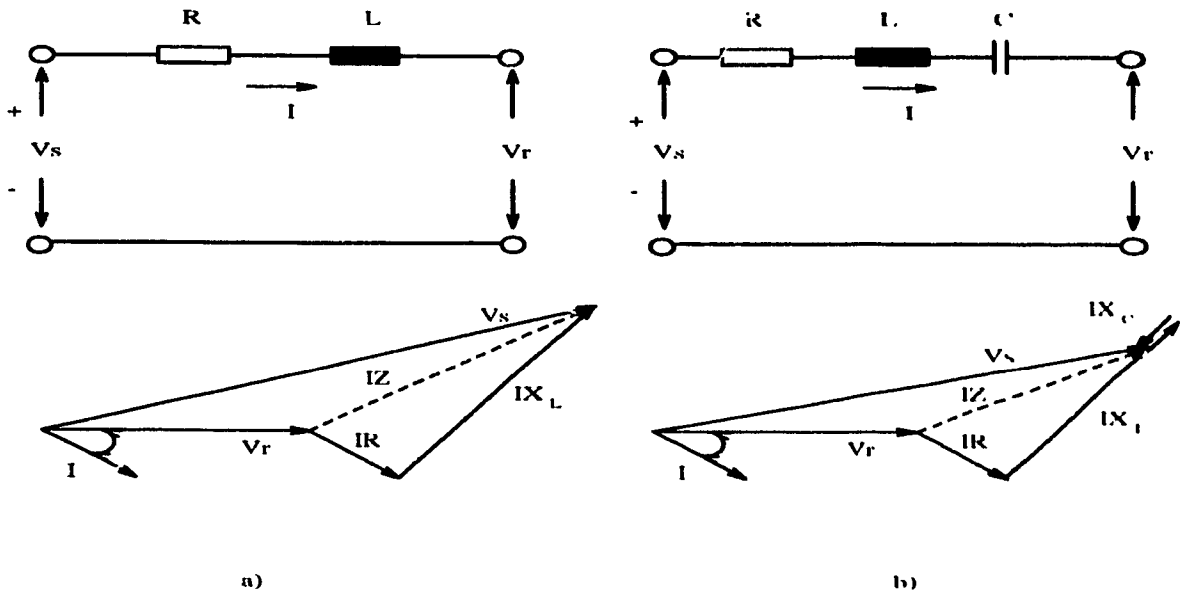

Figure 1.1: Voltage-phasor diagram for a circuit: a) with and b) without series capacitor .

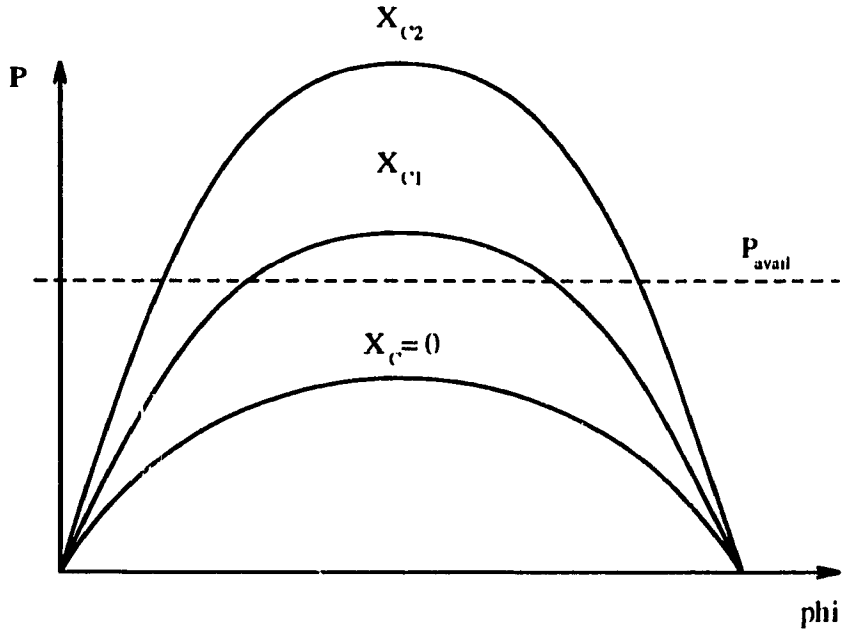

Figure 1.2: Transmitting power as a function of ϕ .

It is well known that switching a transformer/reactor onto a transmission network causes an offset-saturation type of effect in adjacent magnetic devices. This may produce a full spectrum of harmonics along with overvoltages and severe distortion of the fundamental ac waveform. The difference with a series compensated ac line is that the harmonic current injected by the magnetic devices can be much greater, and it has a pulsating characteristic. This is due to the shunt-reactor resonance mode. This behavior is also observed following major ac faults [5].

Thus, the waveforms are more distorted in series compensated transmission lines, and the protection schemes that rely on these waveforms are not as efficient as in the uncompensated transmission lines.

1.3.1 Distance Relays Protection

A distance relay is a device that indicates the position of the fault [3, 2]. The impedance of the transmission line is normally proportional to its length. By measuring the impedance, the distance to the fault position can be estimated. However, it operates only for faults taking place between the relay location and the faulted point. Also, it discriminates between faults to determine whether the fault lies within or outside its operating zone.

In Fig. 1.3 the relay is located at position R. The line current I_R, and the bus

voltage V_R are the two inputs to the relay. Z_S and Z_L are the lumped equivalent impedances of the source and line respectively. The voltage across the relay is given by

$$V_R = I_R Z_L \tag{1.4}$$

where I_R is

$$I_R = \frac{V}{Z_S + Z_L} \tag{1.5}$$

Substitution of eq. 1.5 in eq. 1.4 gives

$$V_R = \frac{1}{(Z_S/Z_L) + 1} V \tag{1.6}$$

Equation 1.6 gives the relationship between the voltage across the relay and the impedance Z_L of the line.

The series capacitor introduces a discontinuity in the impedance of the line, therefore equation 1.6 changes and the values of V_R do not vary linearly. When a line fault occurs near the capacitors, there is a voltage inversion, or at least significant phase shift for relays connected to the line end. The voltage inversion entails loss of directionality for some relays.

The presence of the series capacitor decreases the protection zone of the distance relays. The relay 1 (Fig. 1.4 a) is applied at bus G to protect the line Z_{LR}, but will not detect the faults either in the capacitor or on the line for a distance of X_C. However, the relay 3 will be able to detect the fault that occurs in this line section and incorrect relay operation will occur.

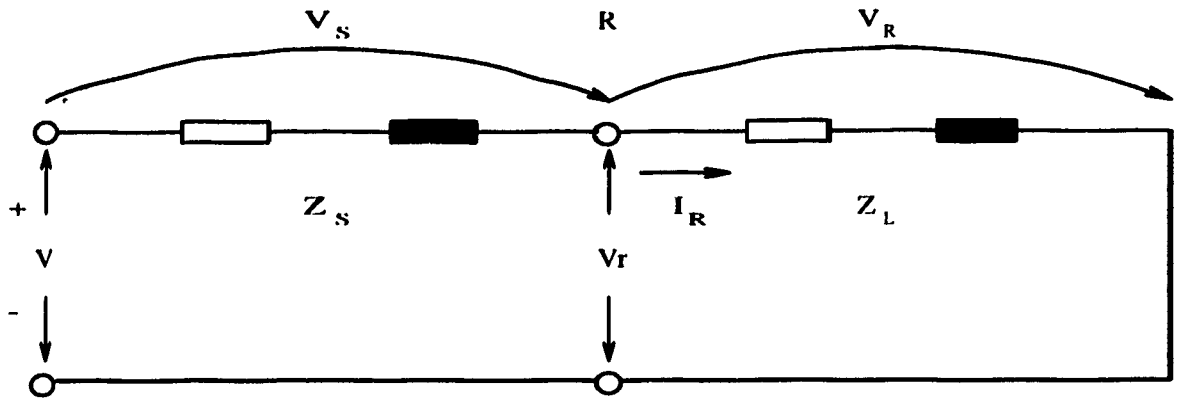

Figure 1.3: Circuit to study relation between source and relay voltage .

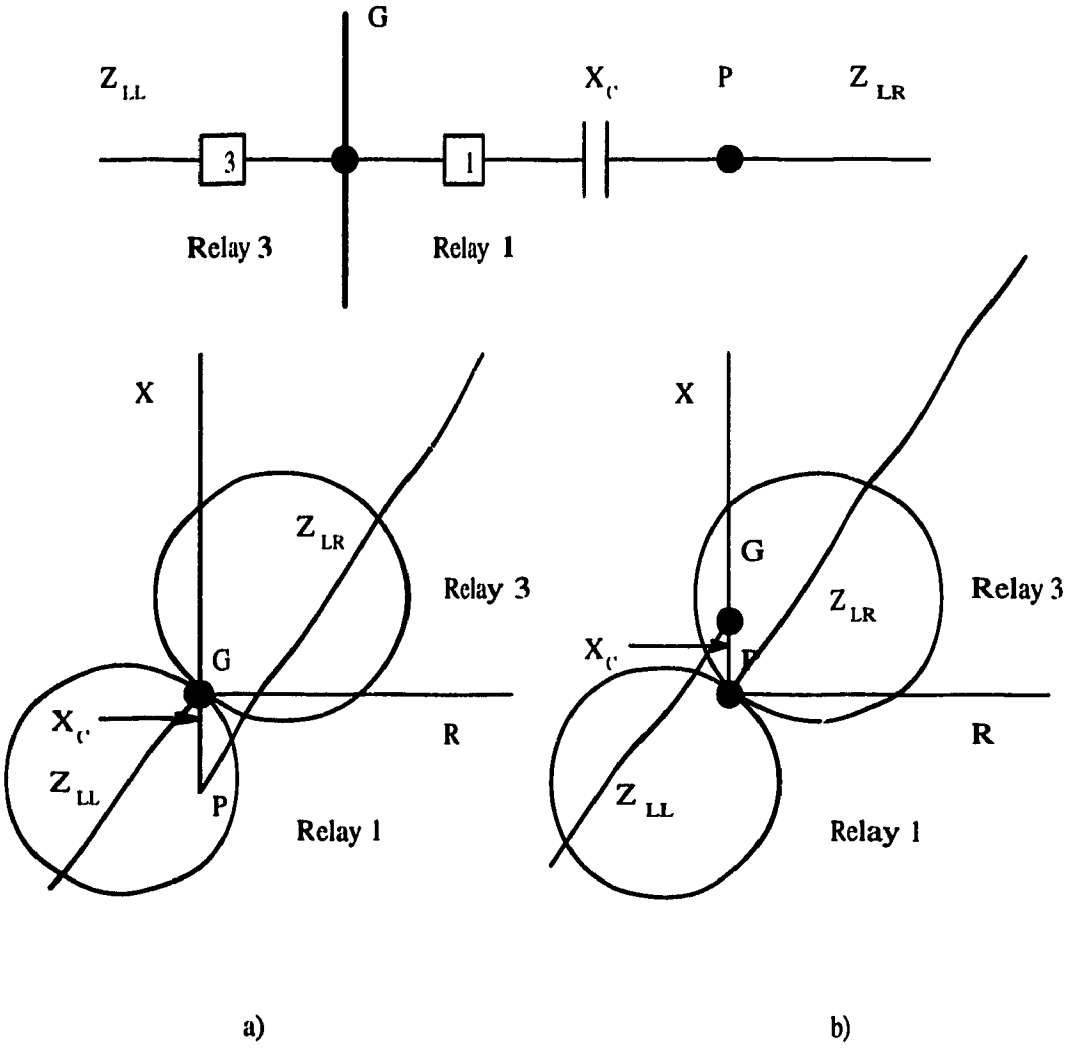

Figure 1.4: Distance relays on a series compensated line .

Changing the location of the relay from point G to P after the capacitor does not solve the problem (Fig. 1.4 b). The relay 1 operates for line Z_{LR}, but also operates for a fault on line Z_{LL} for a distance X_C (Fig. 1.4 b).

The bus voltage exceeds a maximum level set for the series capacitors during the fault periods. The protection of the capacitor is accomplished by connecting spark gaps in parallel with the terminals of the series capacitors. The gaps short the capacitors out when the voltage level exceeds a maximum value set slightly above their insulation capabilities. This moves up the point P and line Z_{LR} to G (Fig. 1.4 a), or the point G and line Z_{LL} moves down to P (Fig. 1.4 b). Although the directional sensing is correct, the protection zone of relay shortens for line Z_{LR} (Fig. 1.4 a), or Z_{LL} (Fig. 1.4 b).

There is always the possibility that the spark gaps will not operate and not short out the capacitor bank. Consequently, incorrect operations of the relays might occur due to distortion of the voltage and current patterns. Therefore, more robust techniques are needed .

1.3.2 Ultra-High Speed (UHS) Protection

In this method [6, 4] the location of the fault can be determined by measuring the time between the departure and arrival of travelling waves at the relaying point

[6, 4]. In Fig. 1.5, the over-voltage due to the fault, V_{s1} is detected first followed by subsequent reflections V_{s2}, and V_{s3}. The time between the detection of V_{s1} and V_{s2} is the time necessary for the waves to propagate to the fault location and back. Therefore, the distance x to the fault from the relay location is given by

$$x = u_p t_1 \tag{1.7}$$

where u_p is the velocity of wave propagation along the line. However, the presence of other reflections, i.e. signals such as V_{s3}, could confuse the relay. Thus, it is very important to develop an algorithm to distinguish the waves of the same type as V_{s1} from other ones.

The values of currents and voltages at the relay location are stored continuously for a cycle. Knowing the signal V_{s1}, the voltage at the fault location would be given as a function of the fault resistance R_f. The voltage at the relay location can be determined as a function of time t_1, which is necessary for waves to propagate to the fault location. Therefore, the relation between the voltage in a no fault condition and the voltages in a fault condition can be expressed by an implicit function depending on time t_1 and R_f,

$$F(t_1, R_f) = 0 \tag{1.8}$$

The protection algorithm is described below:

- parameters of function defined in eq. 1.8 are determined from the prefault conditions,

- the voltage component V_{r1} which would be reflected is calculated,
- the time of arrival, $2l_n$, and incident voltage V_n of any subsequent disturbance is obtained,
- two estimates R_{f1}, and R_{f2} of the actual fault resistance are obtained,
- the two estimates R_{f1}, and R_{f2} would agree only for the primary pulse. This will discriminate between primary pulses and other disturbances.

For a compensated line, the scheme is the same as that shown in Fig. 1.5, but a series capacitor is placed in the middle of the line. The protection algorithm is changed to suit this new configuration. For the first half of the line (Fig. 1.6) between the relaying point P_0 and the capacitor P_2, the equations derived for the untransposed line are used without modifications. For the second half of the line, from just before the series capacitor P_3 to the remote end P_x, the presence of the capacitor has to be considered. The set of equations is changed accordingly.

The algorithm begins with the assumption that the fault occurred in the near half of the line. If no relevant pick is received during the line transit time, it is assumed that the fault may be on the remote half of the line. Then the modified set of equations is used to estimate the location of the fault.

Once again it is obvious that a normally efficient protection scheme for an untransposed transmission lines needs to be adapted for series compensated transmission

lines. Further research in this area is ongoing.

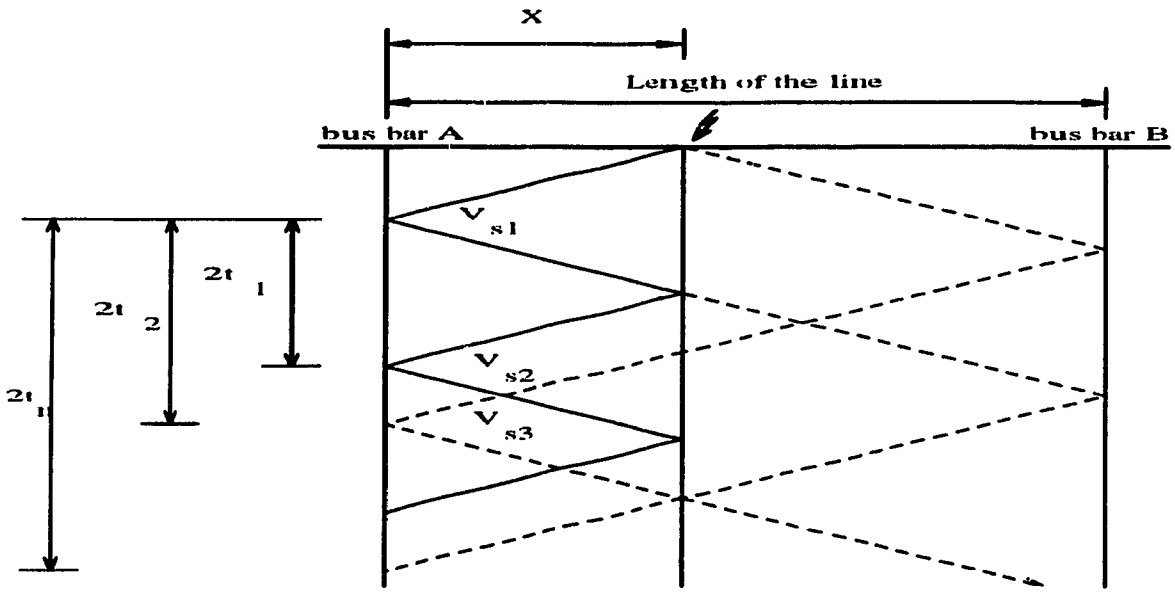

Figure 1.5: Lattice diagram for travelling waves .

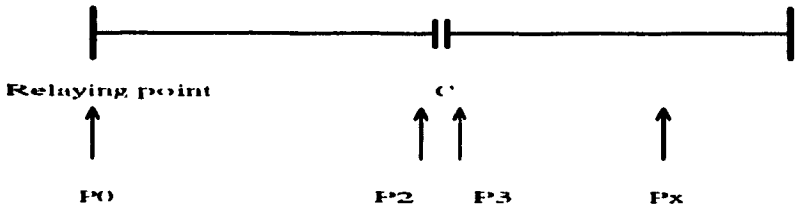

Figure 1.6: Series compensated transmission line.

Chapter 2

Artificial Neural Networks

2.1 Biological Motivation

2.1.1 Single-Neuron Physiology

A neuron [16] is the smallest unit for processing information in a brain. The biological neuron's structure and functions inspired the model of the artificial neuron. The biological neuron has a tree of filamentary dendrites that connect the neuron to other neurons placed in its vicinity. The membrane of the neuron separates the intra-cellular plasma from the interstitial fluid external to the cell. The membrane maintains a potential difference between the two and is permeable to certain ionic species.

Sodium ions (Na^+) are transported out of the cell, while potassium ions (K^+) are received inside it. Negative chloride ions are too large to diffuse though the membrane and stay outside the membrane. The result is an equilibrium potential of about 70 to 100 mV. The potential difference across the cell membrane can be reduced by excitatory inputs to the cell.

The output pulse from the neuron propagates down the axon, which ends with a tree of filamentary dendrites. These transmit the output of the neuron to other neurons. The axons themselves are poor conductors and an amplification mechanism transmits the nerve pulse along the axon. This process is a series of depolarizations

that occur at some nodes, called Nodes of Ranvier. These nodes act as repeater sites for the signal.

2.1.2 The Synaptic Junction

Synaptic junction or *synapse* is the connection between two neurons. The communication between the neurons is accomplished by an exchange of substances, called *neurotransmitters*. The presynaptic membrane allows neurotransmitters to be released. At the other end, they diffuse across the junction and join to postsynaptic membrane at certain receptor sites. There, if the influx is of positive species, it will result in a depolarization of the resting potential, called excitatory effect. If the influx is of negative species, a hyperpolarization effects occurs, called inhibitory effect. The effects are local, and they are summed at the axon hillock. An action potential is generated if the sum is greater than a certain threshold.

2.2 The Artificial Neuron

The artificial neuron has been modeled using the knowledge available from a biological neuron. However, the function of the artificial neuron might differ from the biological neuron in a general case. This is further emphasised by the terminology used for

artificial neurons as processing elements (PE), units, or nodes.

2.2.1 Mathematical Model of Artificial Neuron

The Fig. 2.1 shows a model of the artificial neuron [16]. The input vector X consists of a set of elements $X_1, X_2, \ldots, X_N$. This is applied to the network though a weight matrix, and the result is summed. The sum is given by the relation,

$$NET = \sum_{j=1}^{N} x_j w_j \tag{2.1}$$

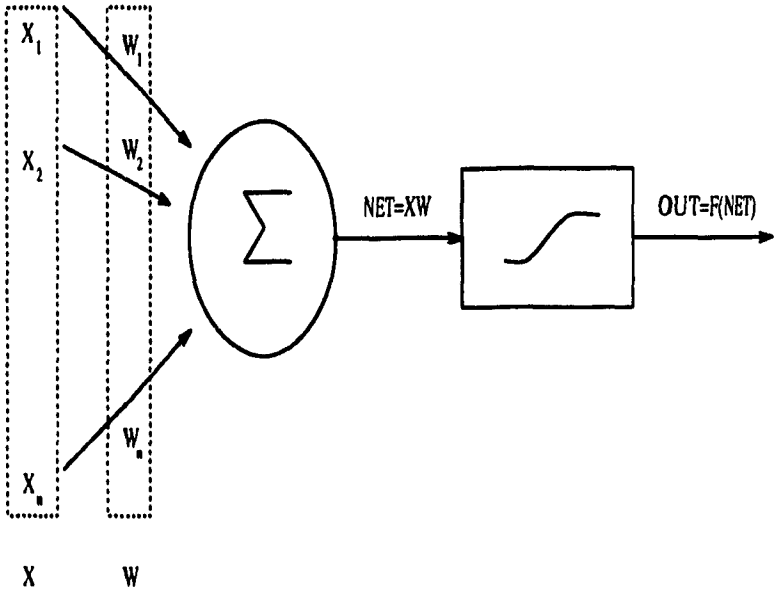

Figure 2.1: The mathematical model for artificial neuron .

where M is the number of output neurons, and N is the number of input elements. The total input NET applied though a function F(X) gives the value of the output of the neuron. The function F is usually the *sigmoid*, or *hyperbolic tangent* function.

Their expressions are shown below, respectively:

$$OUT = 1/(1 + e^{NET}) \tag{2.2}$$

$$OUT = tanh(NET) \tag{2.3}$$

The function F is called the activation function, and has the following properties:

- it saturates the output, such that for large values of the weighted input NET above a certain threshold the increase in output is very small. Thus, the gain is small for a large input,
- for values of input around zero the slope is steep. This provides a large gain for small variations in input around zero. This is usually the case when the training of ANN starts,
- the steepness of the slope around zero can be controlled,
- the expression of the gradient for F with respect to NET, given by eq. 2.2 and 2.3, can be expressed as a function of F itself.

2.2.2 Single-Layer Artificial Neural Networks

Although the neuron itself can perform some tasks, the power of ANNs consist in complex interconnections between neurons. ANNs are hierarchy architectures. Neurons which receive data from the same groups of neurons and send it to a common

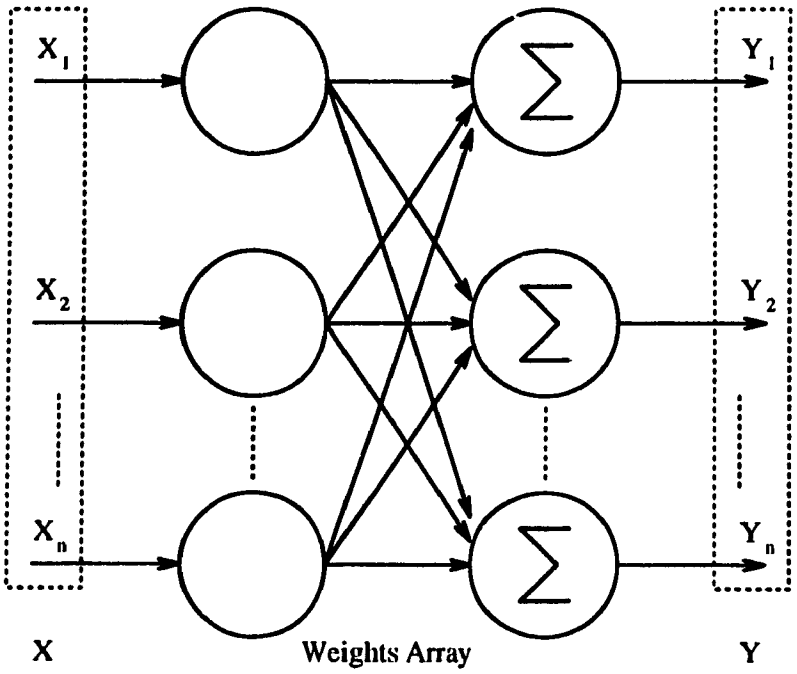

Figure 2.2: Single layer neural network architecture.

group of neurons, form a layer. A single layer ANN is shown in Fig. 2.2. The first layer is a fan-out layer with no processing functions and usually is not considered a layer.

Assume a set of M input vectors, i.e. $X_1, \ldots, X_M$ is fed into the NN, then the NET value described by equation 2.1 can be re-written as a matrix multiplication,

$$NET = XW \tag{2.4}$$

where N is the number of elements of a pattern, and W is the weight matrix.

A single layer ANN can perform more difficult tasks than a single neuron. Con-

sider the activation function as a threshold function,

$$f(x) = \begin{cases} 1, \text{if } x \geq \theta \\ 0, \text{if } x < \theta \end{cases}$$

then a single layer NN can perform pattern classification for patterns that are linearly separable. This condition is very restrictive because many classification problems do not possess linearly separable classes. For such problems the solution is a multi-layer ANN.

2.2.3 Multi-Layer Artificial Neural Networks

A multi-layer ANN is constructed by cascading single-layers of neurons (Fig. 2.3). This will have improved performance only if the activation function of the neurons within a layer is non-linear. Thus, the input/output function of the overall network is non-linear. A multi-layer ANN can classify patterns that are not linearly separable, because the convergence surfaces might have any shape, not only linear.

Having a multi-layer ANN with a linear activation function is pointless, because the NET is a multiplication of matrices, and the following condition holds

$$NET = (XW_1)W_2 = X(W_1W_2)$$

Thus, instead of two layers, and two weight matrices W_1, W_2 one layer NN can be constructed with weight matrix equal to the product of the two matrices W_1W_2.

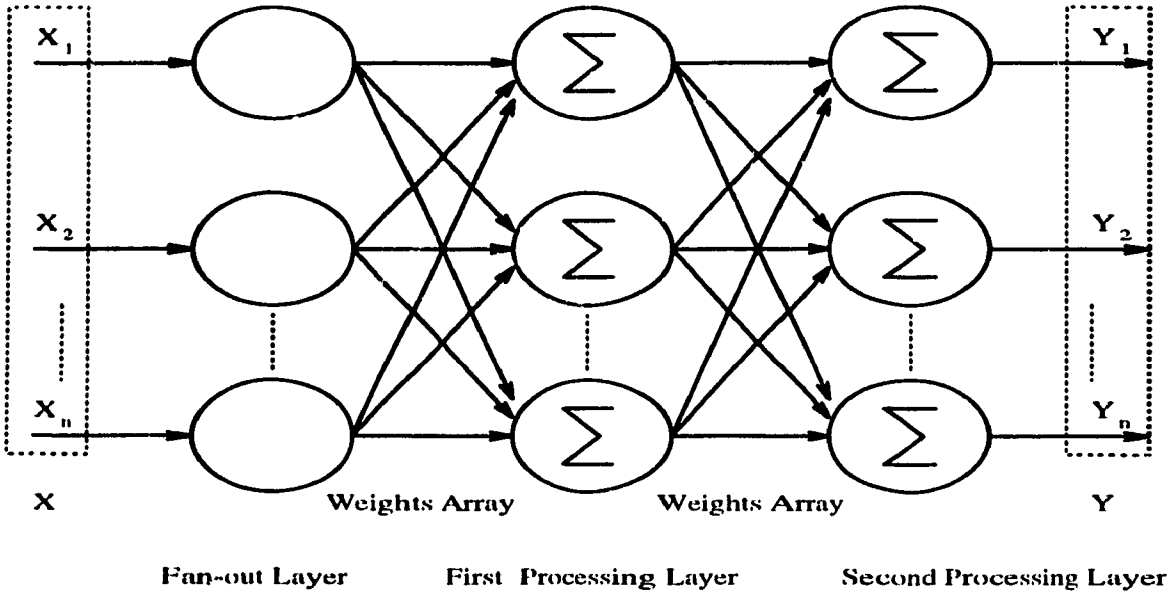

Figure 2.3: Two layers neural network architecture .

2.2.4 Training in Neural Networks

The process of training is simply an algorithm to adjust the weights. During the training, the weights are set to certain values such that the presentation of an input produces the desired output.

In supervised training the desired output is presented with the input. The weights are adjusted such that the error between the desired and actual output decreases to zero.

Unsupervised training consists in presenting input patterns without any other information about the output. The NN extracts the statistical properties of the training set. Based on this, the patterns are classified into different groups.

Also, the training process can be either on-line, or off-line. Off-line means that the NN has been trained before it is used within a system. On-line means that the training is done when the NN is part of a system without any previous knowledge.

2.3 Counter-Propagation Network

Counter-Propagation Network (CPN) was created by Robert Hecht-Nielsen [16, 20]. A CPN works as a "look-up" table, in parallel finding the closest example and reading out its equivalent mapping. Normalized inputs and competition between neurons selects the nearest neighbor.

2.3.1 Counter-Propagation Architecture

CPN is a combination of the competitive Kohonen structure network and Grossberg's *outstar* structure. In this section only a uni-flow version of CPN is discussed (Fig. 2.4).

The first layer of the CPN is an input layer, which pre-process the input vectors. The second layer is a Kohonen layer which consists of *instar* neurons and the third layer is the output layer build from *outstar* neurons.

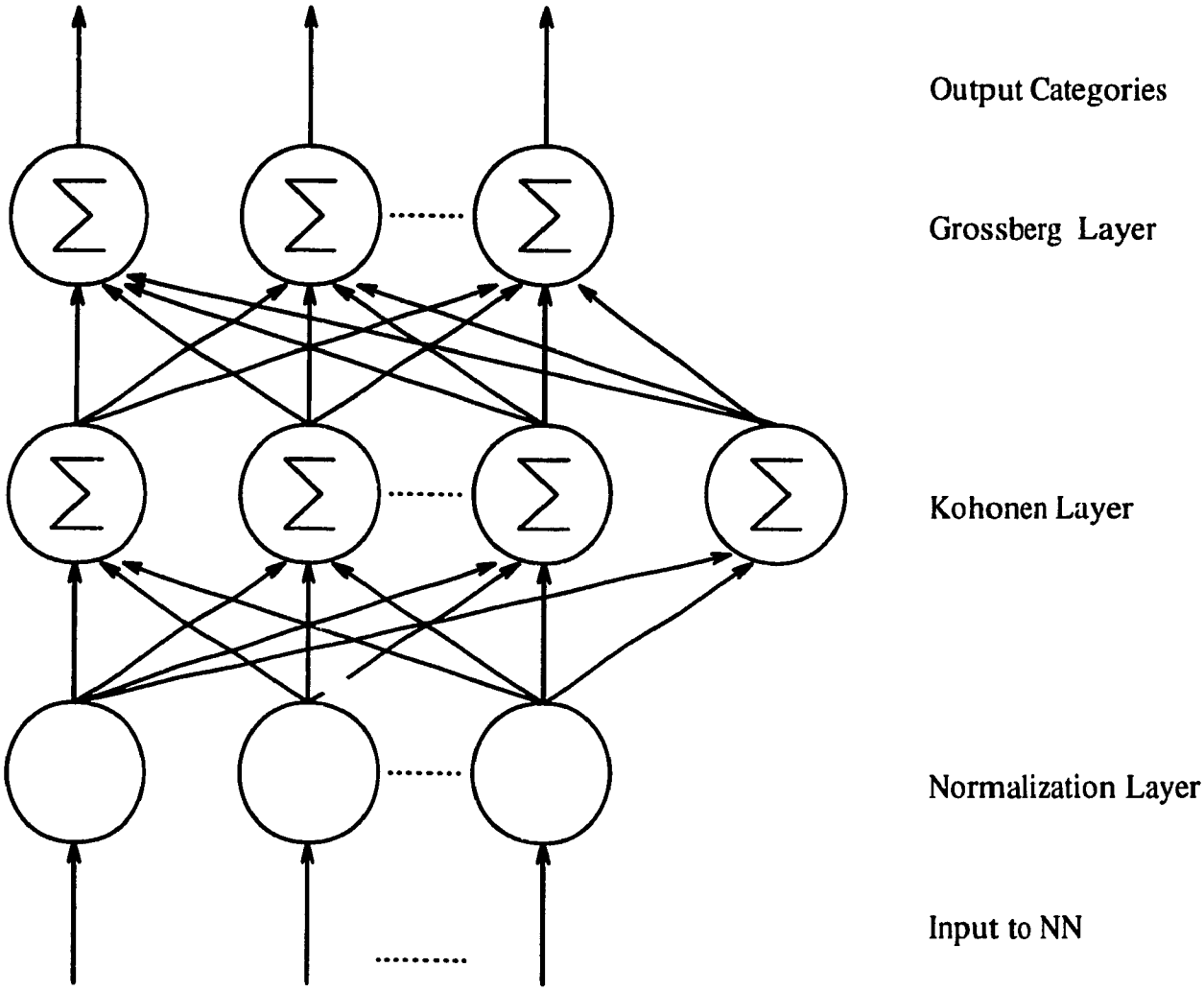

Figure 2.4: CPN architecture .

Two vectors X and Y are applied. The vector Y is the desired output when the input vector X is presented. Applying X as the input vector, after a competition in the Kohonen layer only one neuron fires, while the other neurons are inhibited. The winning neuron activates the connection with the outstar layer selecting in this way a unique output vector. Thus, the uni-flow CPN associates a set of input vectors to a set of output vectors, as a hetero-associative nearest-neighbor classifier.

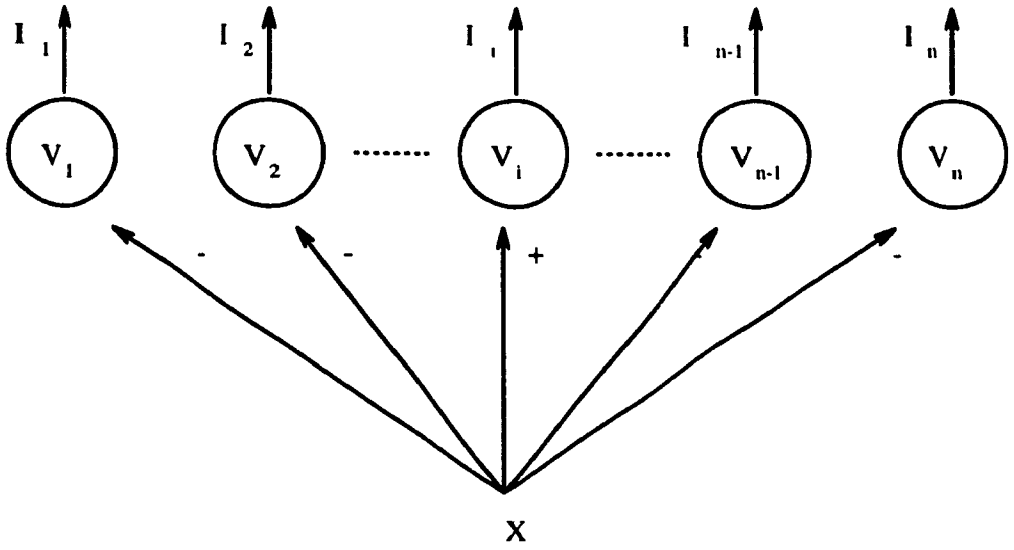

Figure 2.5: The normalization layer, known as *on-center, off-surround* .

The inputs to the Kohonen layer must be normalized to insure the competition among the processing elements (PE) within the layer. The input set of vectors X is normalized using the relation,

$$x_i' = \frac{x_i}{\sqrt{\sum_{j=1}^{N} x_j^2}} \tag{2.5}$$

where x_i, i=1,...,N are components of vector X. Using this normalization all the input vectors lie on the unit sphere. The input vector can either be normalized before is fed into the NN using the pre-processing method described above, or the input layer can do this.

Fig. 2.5 illustrates the normalization method used by the first layer of CPN. Each input element from the input vector is connected to every neuron from the input layer through to the fan-out layer. The connection is positive only for the corresponding neuron in the input layer, the other connections are negative (Fig. 2.5). A quantity

Θ_i is defined for each X_i,

$$\Theta_i = X_i(\sum_i X_i)^{-1} \tag{2.6}$$

The vector $(\Theta_1, \ldots, \Theta_n)^t$ has the property that $\sum_i \Theta_i = 1$.

The differential equation for the output of the layer has the form,

$$\dot{I}_i = -AI_i + (B - I_i)X_i - I_i \sum_{k \neq i} I_i \tag{2.7}$$

where $0 < I_i < B$, and A,B > 0. At equilibrium, the output state is

$$I_i^{eq} = \Theta_i \frac{BX}{A + X} \tag{2.8}$$

where Θ_i is defined by eq. 2.6. The output pattern is normalized, because

$$\sum_i I_i = \frac{BX}{A + X} = B(\frac{A}{X} + 1)^{-1} < B$$

The competitive layer is built up using an *instar* PE. Before the competition the output of one PE is the inner product between the input to the layer and its connection. Let O be the output of this layer, then

$$O = \vec{W} \cdot \vec{I} = \|\vec{W}\| \cdot \|\vec{I}\| \cdot \cos(\theta) \tag{2.9}$$

where θ is the angle between $\vec{W}$, and $\vec{I}$. Since the weights and input vector are normalized to a length 1, the output is given by equation 2.9 becomes

$$O = \cos(\theta) \tag{2.10}$$

The PE with the largest output wins the competition. From 2.10 the output is larger as the angle θ between $\vec{W}$, and $\vec{I}$ is smaller. Thus, the PE, which has the weights closer to the input vector, is the winner. The weights are updated using the formula

$$w(t+1) = w(t) + \alpha(I - w(t))$$

where $w(t+1)$ −states of the weights at next time instant

$w(t)$ −current value of the weights

I −input

α −learning constant, $\alpha < 1$

The output of the winner neuron from the Kohonen layer is input to the next layer (Grossberg layer). During the training process, the outstar PEs from Grossberg layer have also as inputs the set of vectors Y. The weights are set to minimize the error between the desired output Y and actual output. The PEs will respond with pattern Y, every time the PE from the Kohonen layer associated to it wins the competition. After training, the pattern Y is selected, when its correspondent fires, without being presented at the same time with the input vector X. We can say that the CPN learned the associative map between the input and output vectors.

The two structures *instar*, and *outstar* complement each other. The instar recognizes an input pattern and classifies it, and the outstar identifies or names the selected class.

2.3.2 Training of Counter-propagation Network

It is a good practice to use two separate training sets, one for instar PEs, and another one for outstar PEs. First the instar PEs are trained. One PE fires for each cluster of vectors. The input vectors from the training set have to be chosen adequately to represent that cluster. Once the weights of the competitive layer are satisfactorily set, training of the outstar layer can occur.

If all the vectors of one cluster are mapped to the same output vector then the training assigns the weight vector, on the appropriate connections to the output layer, to be equal to the desired output vector. In the case when the vectors are mapped to different output vectors, the weights are set such that the average of those output vectors is reproduced.

2.4 Adaptive Resonance Theory Network - ART2

One of the shortcomings of the CPN is the need for a new training process with changing system conditions [16]. This means that each time new categories of patterns need to be added, or a new pattern added to a category, a new training set has to be generated. This situation is called *stability-plasticity dilemma* and it is valid for most NN architectures. ART tries to solve this dilemma, by using a feedback

mechanism between competitive layer and the input layer of the network.

The information is sent back and forth between the two layers. When the proper patterns develop, a stable oscillation occurs. This is the NN equivalent of resonance. Since the time required for changes in the PE weights is longer than the time that it takes the network to achieve resonance, learning occurs only during the resonant state.

The training of ART is unsupervised. It detects though the orientation system if there is a match with the previous stored patterns, and if not it creates new categories.

2.4.1 ART2 Architecture

Two main types of ART networks are available, ART1, and ART2. The first one has as input binary patterns, while the latter one can have analog patterns as well. The architecture of ART2 is more complex than that of ART1, but the basic principles are the same.

Both have an attentional subsystem and an orienting subsystem. The attentional subsystem consists of two layers of processing elements, F_1 and F_2, and a gain control system. The orienting subsystem is responsible for sensing mismatches between bottom-up and top-down patterns of the F_1 layer.

The F_1 layer is split into six sublayers containing both feedforward and feedback connections. The activity on each sublayer is governed by the equation,

$$\epsilon \dot{x}_k = -Ax_k + (1 - Bx_k)J_k^+ - (C + Dx_k)J_k^- \tag{2.11}$$

where A, B, C and D are constants, J_k^+ and J_k^- represent net excitatory and inhibitory factors. If B and C are set to zero then the asymptotic solution is given by,

$$x_k = \frac{J^+}{A + DJ_k^-} \tag{2.12}$$

There are different values for A, D, J^+, and J_k^- for each of the six sublayers which provide the following equations:

$$w_i = I_i + au_i \tag{2.13}$$

$$x_i = \frac{w_i}{e + \|\vec{w}\|} \tag{2.14}$$

$$v_i = f(x_i) + bf(q_i) \tag{2.15}$$

$$u_i = \frac{v_i}{e + \|\vec{v}\|} \tag{2.16}$$

$$p_i = u_i + \sum_j g(y_i)z_{ij} \tag{2.17}$$

$$q_i = \frac{p_i}{e + \|\vec{p}\|} \tag{2.18}$$

where f is a sigmoid function or

$$f(x) = \begin{cases} 0 & \text{, if } 0 \leq x \leq \theta \\ x & \text{, if } x > \theta \end{cases}$$

with θ positive constant and less than one.

In the processing layer F_2, bottom-up inputs are calculated using the relation:

$$T_j = \sum_i p_i z_{ij} \tag{2.19}$$

After the competition there is only one node winner. The output function of F_2 has the following form,

$$g(y_i) = \begin{cases} d\,T_j = max_k(T_k) & \text{for any k} \\ 0 & \text{otherwise} \end{cases}$$

The equations for Long Term Memory (LTM) have the same form for both bottom-up and top-down weights. Thus the bottom-up weights between node i on layer F_1 and node j on layer F_2 are calculated as

$$\dot{z}_{ji} = g(y_j)(p_i - z_{ji}) \tag{2.20}$$

and top-down weights from node j on layer F_2 to node i on layer F_1 as

$$\dot{z}_{ij} = g(y_j)(p_i - z_{ij}) \tag{2.21}$$

The equation for the orienting subsystem is based on the solution of 2.12

$$r_i = \frac{u_i + cp_i}{\|\vec{u}\| + \|\vec{p}\|} \tag{2.22}$$

where e is assumed equal to zero. The condition for the reset is

$$\frac{\rho}{\|\vec{r}\|} > 1 \tag{2.23}$$

In equation 2.22 the outputs of two sublayers of F_1 participate. The activity of nodes of $\vec{p}$ layer are changing as top-down weights change, but those of $\vec{u}$ layer remain

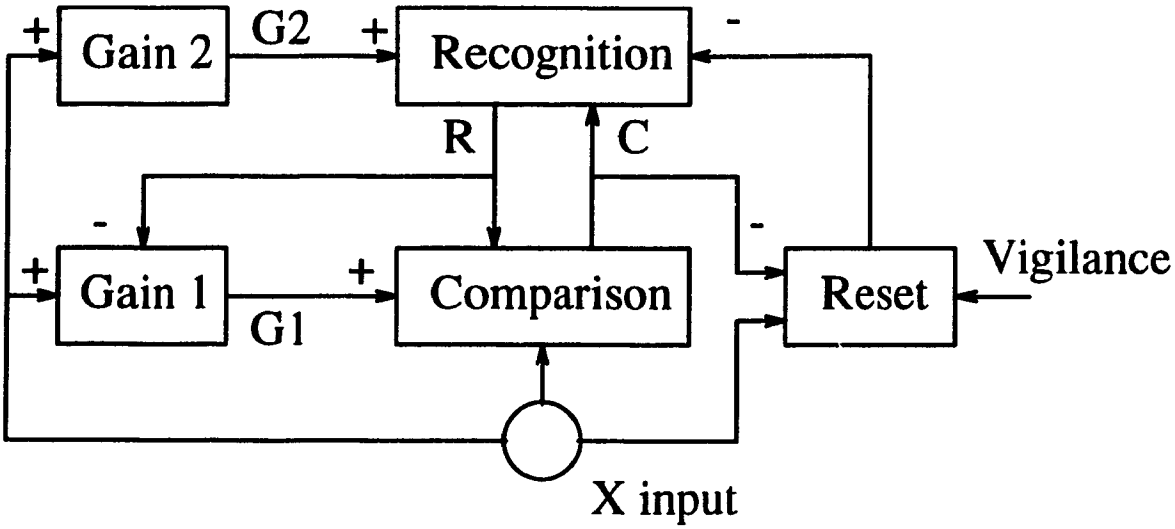

Figure 2.6: ART Architecture .

constant. This prevents a reset while a new pattern is learned. The norm of $\vec{r}$ is given by the expression

$$\|\vec{r}\| = \frac{(1 + 2\|c\vec{p}\| \cos(\vec{u}, \vec{p}) + \|c\vec{p}\|^2)^{1/2}}{1 + \|c\vec{p}\|} \tag{2.24}$$

If $\vec{u}$, and $\vec{p}$ are parallel then $\|\vec{r}\| = 1$, and there is no reset. This is the case when a new pattern needs to be encoded, or no output is received from F_2. A sufficient mismatch between the bottom-up input vector and top-down template will cause a reset.

Chapter 3

Protection of Series Compensated Transmission Lines Using Neural Networks

3.1 Introduction

The use of NNs for protection of transmission lines has been proposed by other authors [7, 10, 12, 13, 14]. The capability of a specifically designed architecture of NN to perform pattern recognition in domains such as speech recognition, image processing, etc. suggested the possibility of using them for fault identification.

Khaparde et al [7] investigated a protective scheme using ADAptive LINEar (ADALINE). In their experiments, they replaced the distance relay in the simplest form with a NN of ADALINE type. The NN could locate the operating point correctly in the decision space. The time requirements to get a decision was comparable with that of conventional relays, without implementation of parallel processing techniques. The limitation of the model proposed in this paper refers to the fact that ADALINE network is trained to take only one decisions based on the values of the bus voltages and line currents. The decision is to detect if the operating point is in the tripping zone or not.

Sobajic and Pao [14] and Aggoune et al [10] used the NNs for power system contingency screening. These authors proposed solutions to assist the operator to take measures to prevent a deteriorate service quality. They analyze the system load flow and were not concerned about fault identification and fault location problems.

In this chapter, a novel protective scheme based on a NN using the information

available from only one end of the transmission line is presented to protect a series compensated line. The NN is placed at the sending-end bus, and its inputs consist of the three bus voltages and line currents. Based on these measurements the NN is required to identify the fault type and fault location. Time domain simulations are performed with two different models for the transmission line and two different NN architectures.

3.2 Systems Modeled

3.2.1 The Transmission Line Model

The 214 km long transmission line is a 735 kV medium-length line (Fig. 3.1). The active power transmitted is 1000 MW and the reactive power is 500 MVAR. The line is modelled with four Π sections (Fig. 3.2) and it is series compensated. The series capacitor (170 uF) is placed in the middle of the transmission line.

The capacitor is protected by a surge arrestor (MOV) of the metal oxide type. The air gap is usually triggered by a special circuit (not shown) to protect and bypass the MOV and capacitor if the absorbed energy exceeds a specified value [17]. A simple two-step non-linear characteristic is used for the arrestor model. The arrestor knee level is 1.5x17.676 kV.

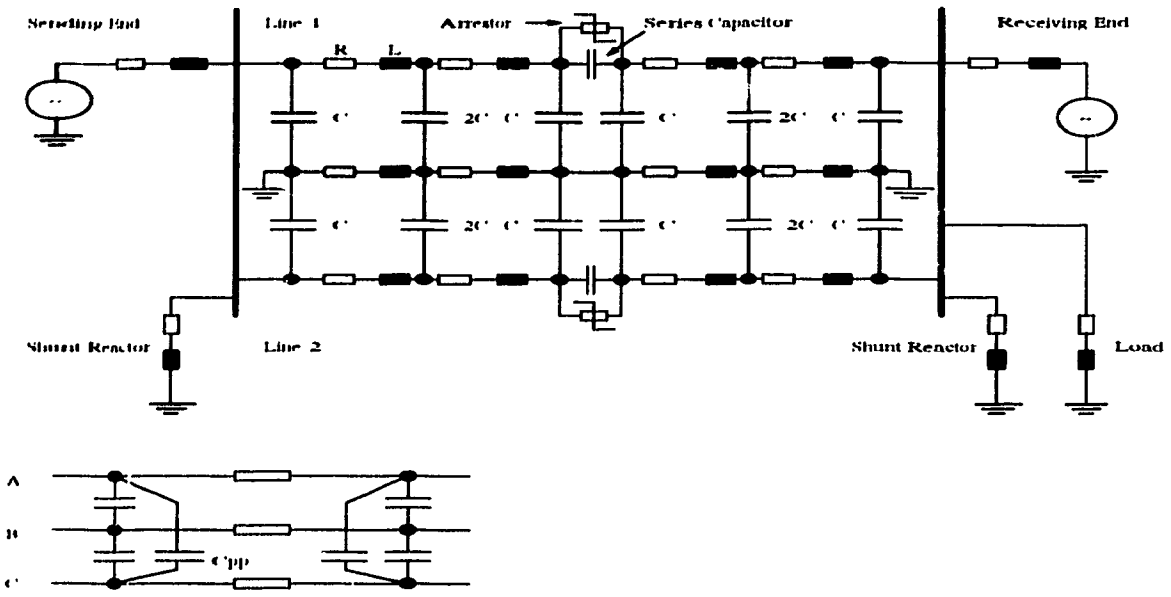

Figure 3.1: Model of the transmission system.

The expression of the voltage at the sending end, function of the voltage and current at the receiving end for a Π sections (Fig. 3.2), is given by,

$$\mathbf{V}_S = (\mathbf{V}_R \times \frac{1}{2}\mathbf{Y} + \mathbf{I}_R)\mathbf{Z} + \mathbf{V}_R$$

$$\mathbf{V}_S = \underbrace{(1 + \frac{1}{2}\mathbf{YZ})}_{\mathbf{A}}\mathbf{V}_R + \underbrace{\mathbf{Z}}_{\mathbf{B}}\mathbf{I}_R \tag{3.1}$$

And the expression of the current at the sending end is

$$\mathbf{I}_S = [(1 + \frac{1}{2}\mathbf{YZ})\mathbf{V}_R + \mathbf{Z}\mathbf{I}_R]\frac{1}{2}\mathbf{Y} + \frac{1}{2}\mathbf{Y} \times (\mathbf{V}_R + \mathbf{I}_R$$

$$\mathbf{I}_S = \underbrace{\mathbf{Y} + \frac{1}{4}\mathbf{Y}^2\mathbf{Z})}_{C}\mathbf{V}_R + \underbrace{(1 + \frac{1}{2}\mathbf{YZ})}_{D}\mathbf{I}_R \tag{3.2}$$

Equations 3.1 and 3.2 can be written in a more compact form as follows,

$$\begin{bmatrix} \mathbf{V}_S \\ \mathbf{I}_S \end{bmatrix} = \begin{bmatrix} 1 + \frac{1}{2}\mathbf{ZY} & \mathbf{Z} \\ \mathbf{Y} + \frac{1}{4}\mathbf{ZY}^2 & 1 + \frac{1}{2}\mathbf{ZY} \end{bmatrix} \begin{bmatrix} \mathbf{V}_R \\ \mathbf{I}_R \end{bmatrix} \tag{3.3}$$

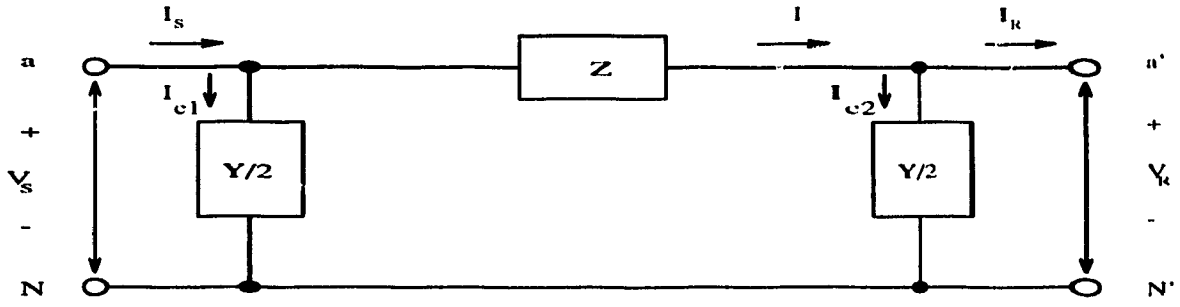

Figure 3.2: T-section equivalent model of transmission line.

The simulations of the power system are done using EMTDC software package [19]. The actual parametric values used for our simulations are listed in the Appendix A. The programs used to run the EMTDC simulation are listed in Appendix B, C and D.

3.2.2 Neural Network Architectures

Counter Propagation Architecture

A general description of the function of CPN is provided in Chapter 2. Here only particular aspects on actual implementation are discussed. CPN is simulated using NeuralWare Profesional II software package [20].

A uni-flow version of CPN is used. The input layer consists of 6 neurons, one for each RMS values of voltage and current of the three phase line. These values are measured at the sending-end, and pre-processed using the following algorithm:

- continuous values of the voltages/currents of each phase (Fig. 3.3 a) are transformed into RMS values (Fig. 3.3 b). The advantage of this transformation is that time varying sinusoidal functions are converted to an almost constant value and the training process is simplified. However, the disadvantage is that a delay of one cycle is introduced to do this transformation,
- the RMS values (Fig. 3.3 b) are scaled in per unit i.e. the voltages/currents are divided by the nominal values respectively. A scaling on different bases is necessary because the nominal values of voltages are different from those for the currents. Therefore, using this procedure a variation in either voltages or currents will have a similar impact in the input vector,

- the pu values of the voltages/currents (Fig. 3.3 b) are normalized using eq. 2.5, and the result is plotted in Fig. 3.3 c. This normalization is done by the CPN in the input layer, as explained in section 2.3.1. Doing this before the signals are fed into the NN makes the computation easier.

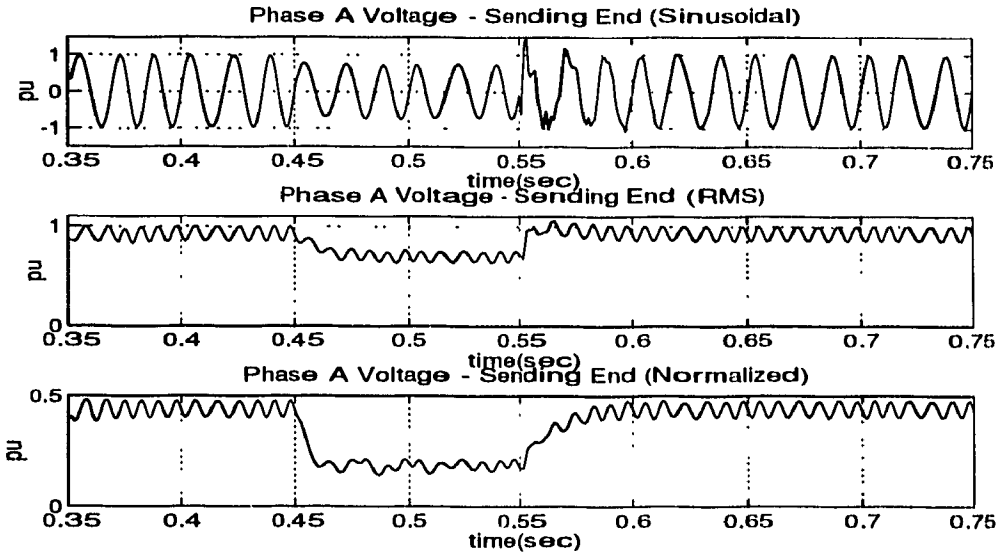

Figure 3.3: Normalization process for phase voltages.

The number of neurons in the hidden layer depends on the number and complexity of the patterns that have to be classified. When the NN is used only for fault identification, the hidden layer has 12 neurons for 11 output categories. When the location of the fault has to be determined also, the number of neurons is increased to 85 for 40 output categories. There are no given rules for these selections, and the number of neurons in the hidden layer was obtained only after practical experimentation. The number of neurons in hidden layer is chosen slightly above the number of categories, because if it is less than the number of categories, the NN will not be

able to classify all categories into distinct classes. While if the number of neurons in the hidden layer is much greater than the number of categories, the convergence process is slower due to the increase in the number of connections.

The number of PE in the hidden layer also depends on the size of the training set. A training set with too many samples of data makes the training process computationally expensive, and a slower convergence, if any, occurs. Typically, for a training set of 1000 samples, 200,000 iterations were necessary for the NN to converge. However, choosing an optimal data set for training the NN is not a trivial task.

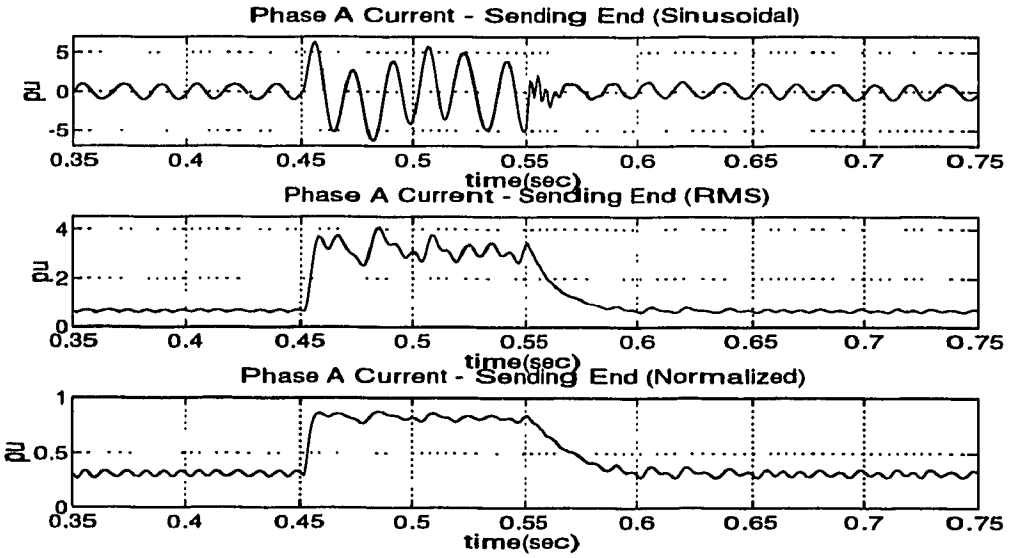

Figure 3.4: Normalization process for phase currents.

In the output layer the number of PEs equals the number of desired categories, hence 11 PEs are needed for the classification of the fault type. The 11 output

categories are listed in Table 3.1.

Table 3.1: Channel No. vs fault type in neural networks figure representation .

Channel No.	Fault Type	
1		No fault
2	LG	AG
3		BG
4		CG
5	DL	AB
6		BC
7		CA
8	DLG	ABG
9		BCG
10		CAG
11	3LG	ABCG

Table 3.2: Channel No. vs fault location in neural networks figure representation .

Channel No.	Location
12	receiving end of the line
13	receiving side of the series capacitor
14	sending side of the series capacitor
15	sending end of the line

Four additional PEs are added to the output layer when the fault location is also

to be considered. In the latter case, the first 11 PEs indicate the fault type and the 4 new added PEs show the location. Possible fault locations are shown in Table 3.2.

ART2 Network

Architecture of ART2 Network was described in Chapter 2. The simulations for ART2 are done using Baoboa software package. The configuration used here has 6 input PEs i.e. one for each three phase voltage and current. The output layer has a maximum of 40 PEs. Since ART is a self organizatory type of network, the number of output PE sets the upper limit for the number of patterns that can be classified.

Many different pre-processing approaches are used for ART2 to investigate the performance of the NN. A common feature for all pre-processing approches is the normalization of the input data. This normalization is required, as in the case of CPN, to equalize changes in anyone particular channel.

The use of the pre-processing method described in section 3.2.2 did not provide better results with ART2 than the CPN. For ART2, better results are obtained by a pre-processing method using Fast Fourier Transformation (FFT). An array is created by appending the normalized continuous values of voltages and currents at every sampling time to the array. The sampling time is 1.0416875 ms, giving a sampling rate of 16 samples a cycle. The array keeps tha data for the last cycle,

hence it has dimensions of 16 by 6 (phase voltages and currents). At every time step, a FFT is calculated for each of the phase voltages and currents signals, and the power spectrum is determined. The magnitude values of the power spectrum are then fed into the NN.

Both pre-processing methods, the one described in section 3.2.2 and the other one using FFT are used in the training of the ART2. The training processes and the discussion of the results are done in the section 3.5.2.

3.3 Fault Identification using CPN

A CPN is placed at the sending end of the transmission line. The particular architecture of the NN for fault identification is presented in section 3.2.2. There are 6 input PEs, 12 PEs in the hidden layer and 11 PEs in the output layer. The input data consists of the RMS values of phase voltages and currents from the sending end. NN identifies 10 types of fault applied at the receiving end and indicates no fault, if there is no fault detected on the line.

The transmission line is modeled using II - sections (see section 3.2.1 for more details on transmission line model). A series capacitor is placed in the middle of the line. Parametric values for different components are presented in Appendix 1.

Two different features are chosen to characterize a fault type:

- the maximum of the normalized values of the voltages and currents,
- the average of the normalized values of the voltages and currents for three cycles.

3.3.1 Training using the maximum values of the input

Before using the ANN to classify faults and faults location, it is necessary to train the ANN. For this a training set is necessary. Before selecting a training set, it is important to consider the characteristics of different patterns and evaluate if these characteristics can uniquely define every particular fault. To select a characteristic feature for a fault, all the vectors consisting of the voltages/currents values at each time step for a cycle need to be looked at. This creates a cluster of vectors that characterize a pattern, i.e. 100 vectors for our time step. Therefore, for 11 patterns, the training set grows to 1,100 entries. The more entries in the training set, the more PEs that are required in the hidden layer. A large number of PEs in the hidden layer makes the convergence of the NN slower.

To reduce the training set and the computation time, only the maximum values over 3 cycles within the fault period of the voltages/currents was used. Thus, there was only one vector that characterized a fault. The size of the training set was thus reduced from 1,100 to 11 (Table 3.3). The convergence is therefore faster (only 4500

iterations compared to 200,000 for a training set of 1,100 entries) and the error during the training decreases rapidly to zero. It remains to be seen if this massive reduction in the training set gives acceptable results during the recall process.

In Fig. 3.5, a single phase fault (AG), double phase fault (AB) and double phase to ground fault (ABG) are applied at 0.45 sec. and cleared at 0.55 sec. There is a delay of one cycle in the detection of the fault due to the computation of RMS values. Furthermore, the end of the fault is detected with a delay due to the presence of harmonics.

For no fault, line to ground, and three phase to ground fault cases there is a stable response, but for double line, and double line to ground fault cases, there are a lot of oscillations, i.e. for cases affecting the same phases as AB, ABG, etc. faults (Figs. 3.5, 3.6). These oscillations are caused by the way the characteristic features for a fault type were chosen. The entries in the training set for double line and double line to ground faults (Table 3.3) are very close, and the NN cannot distinguish between the two faults in all cases.

Therefore, it can be concluded that using the characteristic feature of maximum values over 3 cycles of the voltages/currents within the fault period reduced the training set. This smaller training set improved the convergence speed of the NN. Although these are important gains, an alternative method was investigated to verify if improved the results could be obtained by the recall process.

Table 3.3: Training set using maximum values of input data .

Case	Input Vector						Outputs
	V_a	V_b	V_c	I_a	I_b	I_c	
NF	0.4401	0.4393	0.4361	0.3128	0.3625	0.4401	NF=1
AG	0.2194	0.2098	0.2268	0.2687	0.8232	0.3016	AG=1
BG	0.2355	0.2284	0.2268	0.2687	0.8232	0.3016	BG=1
CG	0.2363	0.2368	0.2364	0.2428	0.2964	0.8278	CG=1
AB	0.1307	0.1351	0.0912	0.7094	0.6590	0.1369	AB=1
BC	0.1025	0.1012	0.0998	0.0859	0.7253	0.6601	BC=1
CA	0.1495	0.0892	0.1599	0.6477	0.0736	0.7205	CA=1
ABG	0.1323	0.1285	0.0926	0.7149	0.6534	0.1389	ABG=1
BCG	0.1004	0.0974	0.0964	0.1157	0.7280	0.6540	BCG=1
CAG	0.1480	0.0883	0.1565	0.6435	0.1064	0.7213	CAG=1
ABCG	0.1261	0.0835	0.0858	0.6024	0.5825	0.5172	ABC=1

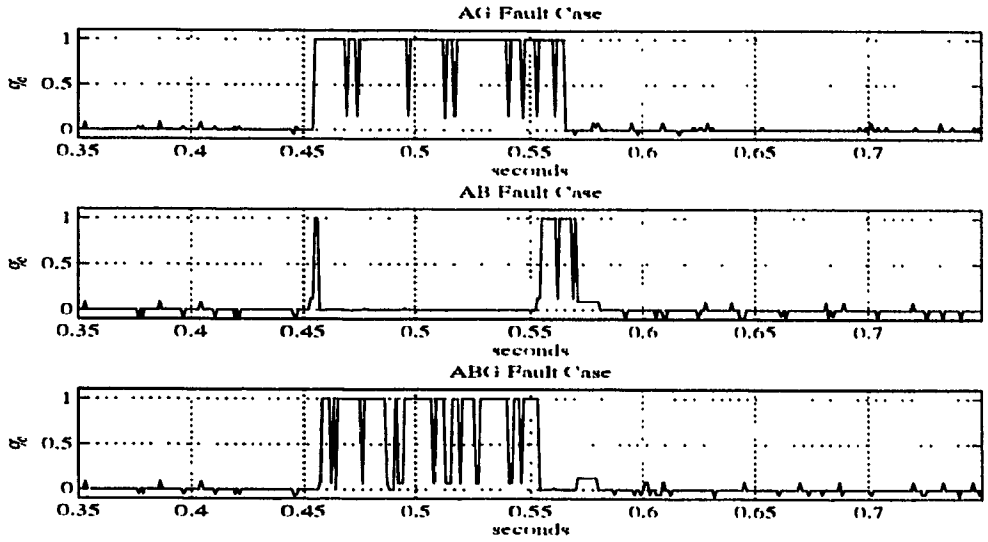

Figure 3.5: NN output for AG, AB, and ABG fault cases - Training set uses maximum values .

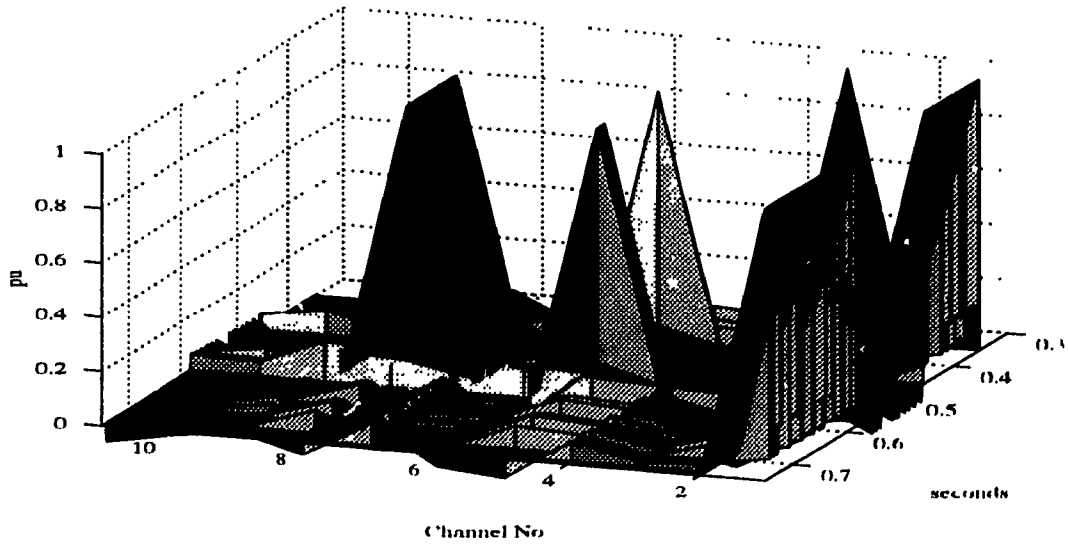

Figure 3.6: Overall view of the NN output for ABG fault - Training set uses maximum values .

3.3.2 Training using the average values of the input

Choosing maximum values as the characteristic features for the fault cases did not result in satisfactory recall. During the recall process, few vectors are in the vicinity of the maximum value characteristic vector and therefore classification errors occur for the input vectors which are not close by. This suggested the use of an *average value* over 3 cycles characteristic vector of the voltages/currents for categorizing a fault.

The new average values training set is given in the Table 3.4.

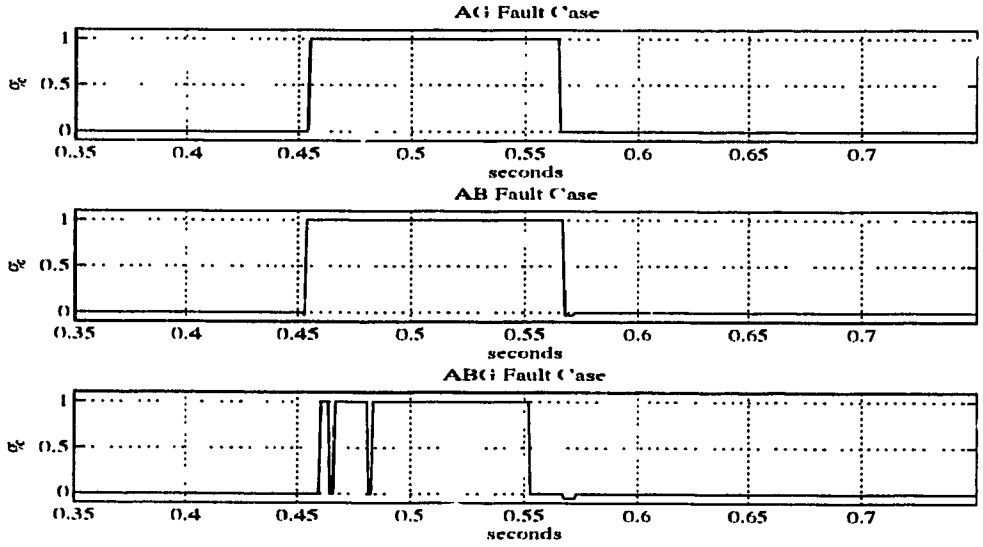

Figure 3.7: NN output for AG, AB, and ABG fault cases. Training set uses average values .

The Fig. 3.7 shows the recall results obtained for the same fault types plotted in

Fig. 3.5. The response for double line and double line to ground is much improved. Indecision regions are much reduced (Fig. 3.8) when compared to the previous results (Fig. 3.6). Similar results were obtained for other types of fault.

Table 3.4: Training set using average values of input data .

Case	Input Vector						Outputs
	V_a	V_b	V_c	I_a	I_b	I_c	
NF	0.4246	0.4214	0.4289	0.3940	0.3872	0.3912	NF=1
AG	0.1574	0.2145	0.2235	0.8739	0.2140	0.2642	AG=1
BG	0.2254	0.1593	0.2222	0.2696	0.8686	0.2174	BG=1
CG	0.2238	0.2275	0.1646	0.2244	0.2699	0.8648	CG=1
AB	0.0874	0.0938	0.1282	0.7316	0.6467	0.1169	AB=1
BC	0.1475	0.1014	0.1101	0.1369	0.7376	0.6270	BC=1
CA	0.0952	0.1289	0.0892	0.6464	0.1184	0.7311	CA=1
ABG	0.0760	0.0831	0.1221	0.7319	0.6460	0.1398	ABG=1
BCG	0.1385	0.0888	0.0980	0.1617	0.7343	0.6309	BCG=1
CAG	0.0850	0.1225	0.0782	0.6525	0.1418	0.7252	ABC=1

The delay of one cycle in the detection of the fault, due to the computation of RMS values, is still present. Moreover, for the case of the ABG fault, the time elapsed until the detection of the beginning of the fault is shorter. In Fig. 3.8, it is observed that the beginning of the fault is detected as a BG fault. Then the NN indicates an AB fault, and later an ABG fault. This sequence of events is in accord with the

theory of fault build up, and it is not an error in the indication provided by the NN. A similar phenomenon is present when the fault is cleared. This method therefore provides good results for fault identification.

In the next section a similar method is proposed to locate the fault. The expanded NN will now have two tasks:

- to identify the type of fault, and
- to locate the position of the fault.

3.4 Fault Identification and Location using CPN

The combined tasks of identification and location of the fault are accomplished by using a CPN. The number of PEs in the hidden layer is increased from 12 to 85, according to the number of the categories to be classified. Now there are 15 PEs in the output layer of the NN. The first 11 PEs (channels 0 to 10) indicate the fault type, while next 4 PEs (channels 11 to 14) indicate the location of the fault. For the moment only four locations of the fault are identifiable: at the receiving end (R), after (A), and before (B) the series capacitor, and at the sending end (S).

The characteristic feature of the fault categories is the average over 3 cycles of

the normalized RMS values of the phase voltages/currents. This method provided very good results previously, and is expected to have the same efficiency for this case. The training set is presented in Table 3.5. Now the fault are applied sequentially to all four locations compared to the previous case, fault identification only, when the faults were applied only to the receiving end.

The results of the recall process are presented in a three-dimensional format for a better appreciation of the overall response of the NN. The cases presented include line to ground (BG), double line(BC), and double line to ground (BCG) faults. Each fault is applied sequentially at all four locations and the results are compared.

The results for the case of line to ground fault (BG) are presented in Figs. 3.9, 3.10, 3.11, and 3.12 for each fault location respectively. The fault type at all the locations is very well classified.

The results of the NN for a line to ground fault (BG) applied at the receiving end of the line are presented in Fig. 3.9. In this case, there is an interval where the NN indicates the fault location as being after the capacitor. This erroneous indication comes shortly after the fault inception and classification as BG. The response switches back to the correct indication after a few iterations. The error is due to the transient response at the beginning of the fault. The overall response of NN is very good, with a fast identification of the fault type, and consistent indication of the fault location.

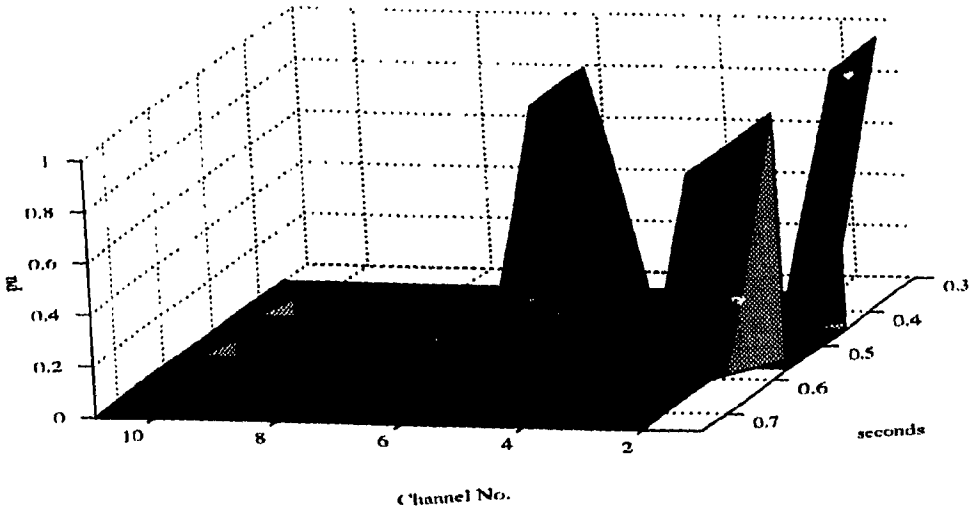

Figure 3.8: Overall view of the NN output for ABG fault. Training set uses average values .

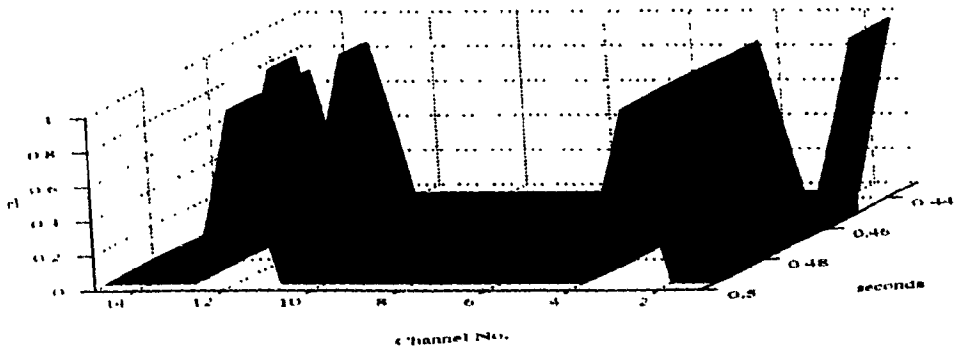

Figure 3.9: Overall view of the NN output for BG fault at the receiving end.

Table 3.5: Training set for fault identification and location .

Case	Input Vector						Outputs
	V_a	V_b	V_c	I_a	I_b	I_c	
NF	0.4230	0.4238	0.4171	0.3924	0.3986	0.3929	NF=1;R=1
AG	0.1619	0.2169	0.2184	0.8742	0.2188	0.2582	AG=1;R=1
BG	0.1985	0.1463	0.1915	0.2337	0.8990	0.1987	BG=1;R=1
CG	0.1880	0.1926	0.1401	0.1916	0.2292	0.9047	CG=1;R=1
AB	0.0886	0.0917	0.1195	0.7273	0.6540	0.1126	AB=1;R=1
BC	0.0833	0.0614	0.0628	0.0773	0.7079	0.6914	BC=1;R=1
CA	0.0903	0.1175	0.0845	0.6628	0.1106	0.7205	CA=1;R=1
ABG	0.0824	0.0881	0.1230	0.6649	0.7126	0.1424	ABG=1;R=1
BCG	0.1097	0.0740	0.0785	0.1262	0.6935	0.6923	BCG=1;R=1
CAG	0.0858	0.1208	0.0804	0.6570	0.1404	0.7211	CAG=1;R=1
ABCG	0.0560	0.0570	0.0569	0.5236	0.5837	0.6127	ABCG=1;R=1
AG	0.1062	0.1719	0.1709	0.9318	0.1608	0.1891	AG=1;A=1
BG	0.1758	0.1072	0.1715	0.1942	0.9292	0.1640	BG=1;A=1
CG	0.1749	0.1772	0.1071	0.1634	0.1973	0.9278	CG=1;A=1
AB	0.0651	0.0668	0.1006	0.7205	0.6730	0.0948	AB=1;A=1
BC	0.1240	0.0802	0.0809	0.1150	0.7350	0.6466	BC=1;A=1
CA	0.0692	0.1047	0.0665	0.6630	0.0985	0.7283	CA=1;A=1
ABG	0.0520	0.0550	0.0960	0.7194	0.6757	0.1040	ABG=1;A=1
BCG	0.1143	0.0613	0.0645	0.1218	0.7367	0.6491	BCG=1;A=1

Table 3.6: Training set for fault identification and location (continued) .

Case	Input Vector						Outputs
	V_a	V_b	V_c	I_a	I_b	I_c	
CAG	0.0552	0.0978	0.0514	0.6726	0.1060	0.7218	CAG=1;A=1
ABCG	0.0372	0.0372	0.0370	0.6314	0.5576	0.5349	ABCG=1;A=1
AG	0.0918	0.1552	0.1547	0.9443	0.1477	0.1728	AG=1;B=1
BG	0.1429	0.0845	0.1392	0.1577	0.9544	0.1314	BG=1;B=1
CG	0.1403	0.1423	0.0840	0.1329	0.1607	0.9536	CG=1;B=1
AB	0.9329	0.0195	0.0297	0.2606	0.2442	0.0279	AB=1;B=1
BC	0.0663	0.0439	0.0443	0.0614	0.7113	0.6942	BC=1;B=1
CA	0.0584	0.0867	0.0552	0.6779	0.0815	0.7209	CA=1;B=1
ABG	0.0434	0.0495	0.0860	0.6701	0.7282	0.0940	ABG=1;B=1
BCG	0.0765	0.0407	0.0459	0.0825	0.6975	0.7049	BCG=1;B=1
CAG	0.0464	0.0813	0.0426	0.6963	0.0886	0.7047	CAG=1;B=1
ABCG	0.0286	0 0303	0.0315	0.5308	0.5797	0.6158	ABCG=1;B=1
AG	0.0110	0.0628	0.0609	0.9928	0.0524	0.0613	AG=1;S=1
BG	0.0765	0.0128	0.0765	0.0755	0.9890	0.0642	BG=1;S=1
CG	0.0804	0.0794	0.0152	0.0664	0.0795	0.9880	CG=1;S=1
AB	0.0278	0.0275	0.0529	0.7152	0.6939	0.0499	AB=1;S=1
BC	0.0717	0.0374	0.0379	0.0665	0.7290	0.6753	BC=1;S=1
CA	0.0290	0.0552	0.0294	0.6876	0.0519	0.7208	CA=1;S=1

Table 3.7: Training set for fault identification and location (continued).

Case	Input Vector						Outputs
	V_a	V_b	V_c	I_a	I_b	I_c	
ABG	0.0086	0.0081	0.0480	0.7771	0.6259	0.0431	ABG=1;S=1
BCG	0.0558	0.0093	0.0106	0.0490	0.7174	0.6924	BCG=1;S=1
CAG	0.0087	0.0494	0.0094	0.7877	0.0441	0.6123	CAG=1;S=1
ABCG	0.0074	0.0069	0.0079	0.6662	0.5363	0.5180	ABCG=1;S=1

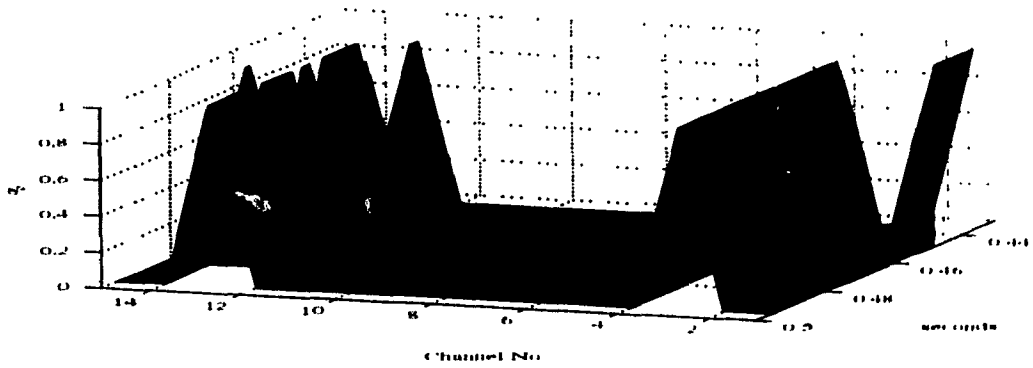

Figure 3.10: Overall view of the NN output for BG fault after capacitor.

The response to a fault applied after the capacitor is plotted in Fig. 3.10. There is an oscillation in the response of NN regarding the fault location. The indication oscillates between the receiving end (the correct position) and after the capacitor due to the impedance of the capacitor. The impedance is too small to make a difference for values of voltages and currents at the sending end. Thus, the two patterns, fault location at receiving end and fault location after the capacitor are not entirely separable, and some oscillations are expected. Also, this is valid when the fault is applied before the capacitor (Fig. 3.11). Although there is an incertainty in locating the fault for short periods, the response is very good.

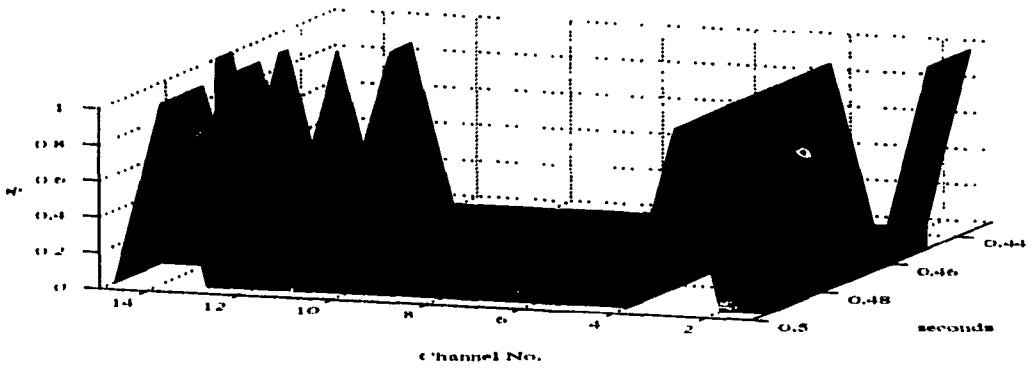

Figure 3.11: Overall view of the NN output for BG fault before capacitor.

The same excellent results are obtained for the case when the fault is applied to the sending end (Fig. 3.12). The NN indicates a few erroneous points at the beginning of the fault, but afterward recovers and provides a good indication. The error at the beginning of the fault is due to the transition period.

In Figs. 3.13, 3.14, 3.15, and 3.16 are shown the results of the NN when a double

line fault (BC) is applied individually to all four locations, respectively.

No fault type indication is indicated several times during the fault period. The NN cannot classify the data in one of the known categories. Several times it indicates a BCG fault due to similar data for double line and double line to ground. Also, there are oscillations in fault location between after and before capacitor, as in the case of a line to ground fault.

Although, the response is not always equal to the desired one, the error is within an acceptable range of 10% within the fault period. A much better response is obtained in identification and location of the faults applied before capacitor (Fig. 3.15), and sending end (Fig. 3.16). In these cases there is a period of uncertainty immediately after the fault application due to transient factors. However, the erroneous indication refers to a double line to ground fault, which is not a disastrous false alarm because at least the correct faulted phases are identified.

The case of a double line to ground fault (BCG) at different locations is presented in the Figs. 3.17, 3.18, 3.19, and 3.20. This type of fault is characterized by much instability. There are more oscillations between the double line (BC) and double line to ground fault (BCG) indication for this case than for the double line fault case. In addition to the oscillations in identification of the fault type, there are oscillations in the indication of the fault location. The causes for uncertainty in fault location are similar to those previously discussed fault cases.

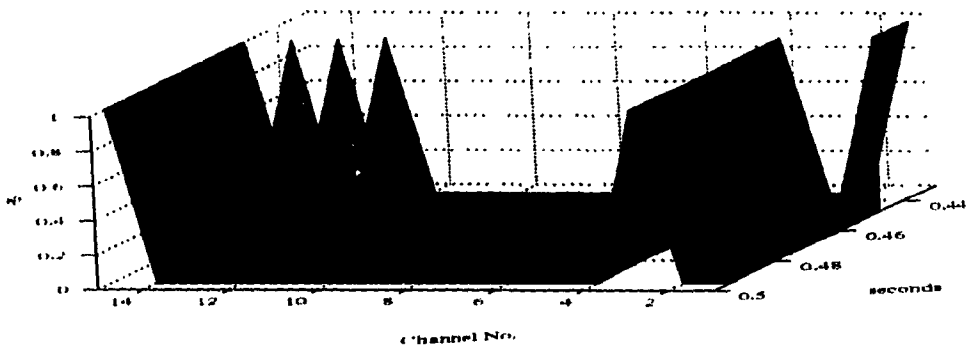

Figure 3.12: Overall view of the NN output for BG fault at sending end.

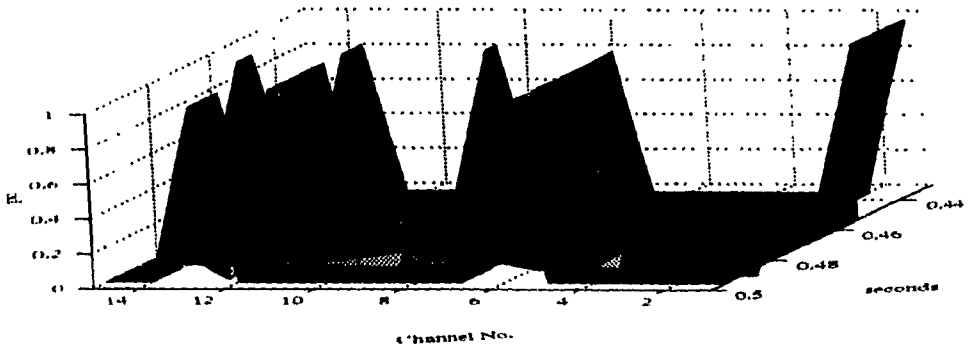

Figure 3.13: Overall view of the NN output for BC fault at the receiving end.

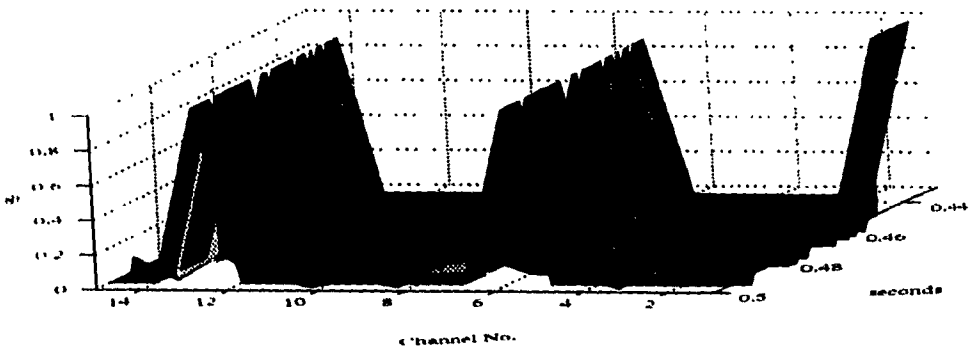

Figure 3.14: Overall view of the NN output for BC fault after capacitor.

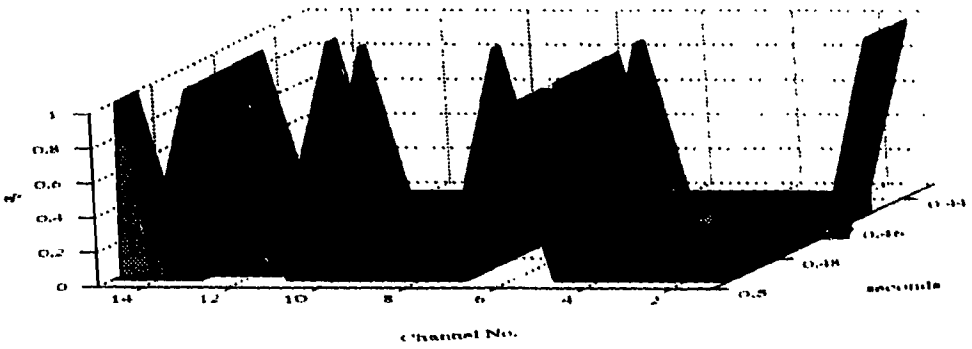

Figure 3.15: Overall view of the NN output for BC fault before capacitor.

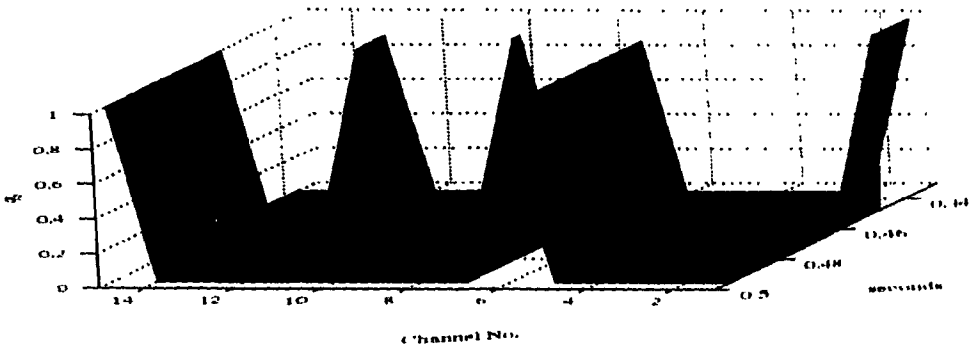

Figure 3.16: Overall view of the NN output for BC fault at sending end.

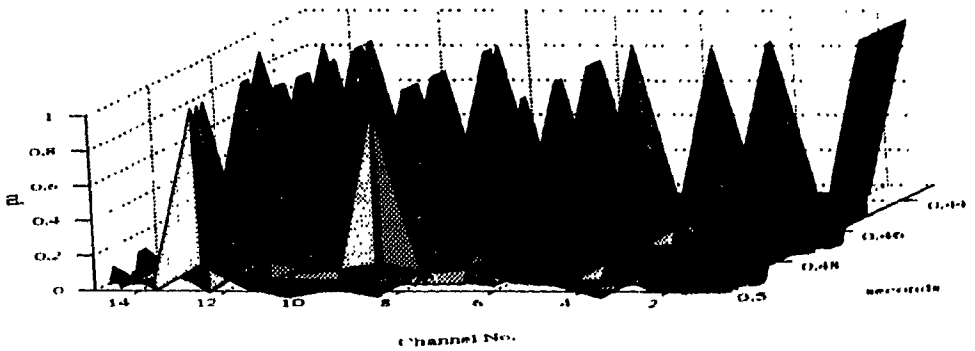

Figure 3.17: Overall view of the NN output for BCG fault at the receiving end.

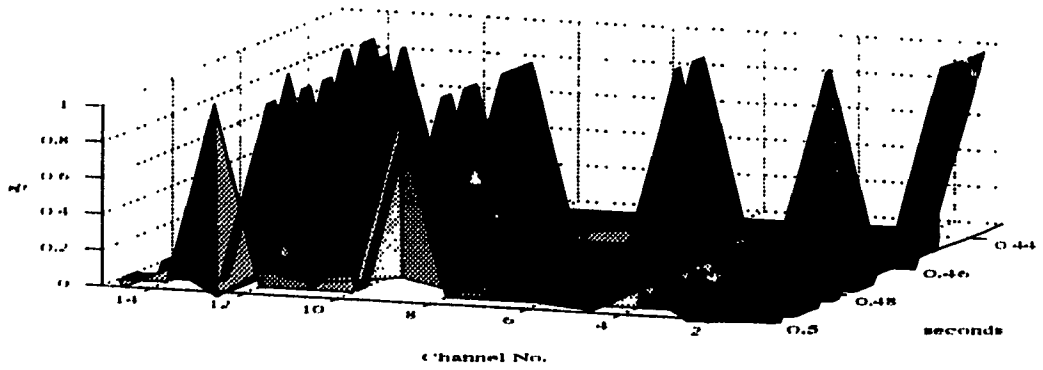

Figure 3.18: Overall view of the NN output for BCG fault after capacitor.

The combined effect of the oscillations caused by fault identification and location make the overall response of the NN to be more confusing.

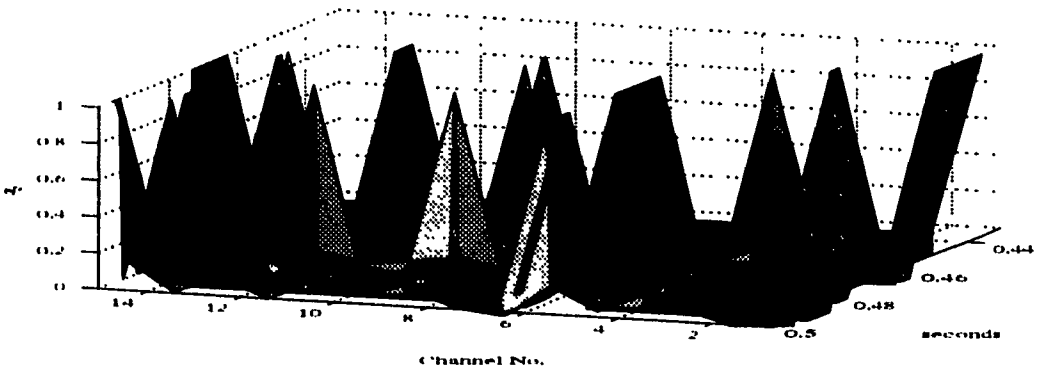

Figure 3.19: Overall view of the NN output for BCG fault before capacitor.

3.5 Fault Identification using ART2

Although the CPN architecture provided very good results in identification and location of the fault it does have one shortcoming. The CPN needs a new training set for

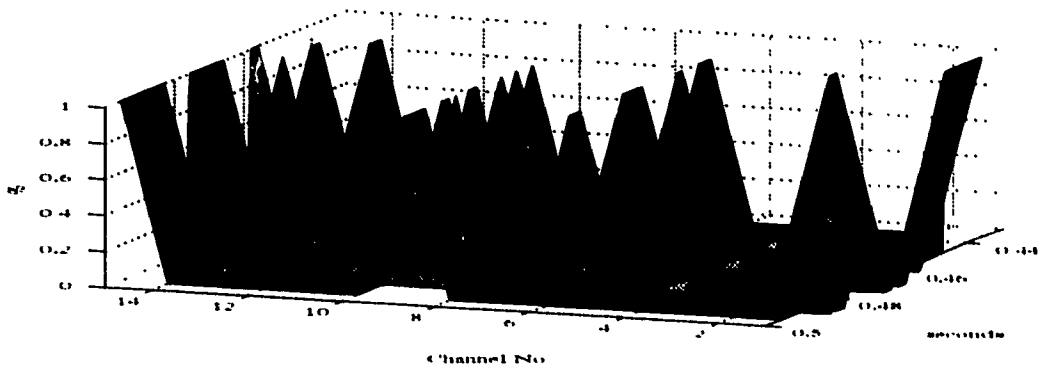

Figure 3.20: Overall view of the NN output for BCG fault at sending end.

every change in the parameters of the system. To overcome this the new ART architecture is used. ART has the plasticity to learn new categories while preserving the information previously acquired (section 2.4). However, there is a defined maximum limit to the new categories that can be learned.

3.5.1 Pre-Processing of Data for ART2

In common with the CPN architecture, the input data used by ART2 for fault identification are the pre-processed phase voltage/current signals available at the sending end bus.

When ART2 was used in a similar manner to CPN, it did not provide better results when the training set used was the average of the normalized RMS values of the voltages and currents. Furthermore, ART2 had additional difficulties to discriminate between double line and double line to ground faults. To improve on the classification

with ART2, a new pre processing procedure based on FFT techniques to monitor signals was used (section 3.2.2).

A matrix is defined such that its rows hold the sinusoidal values of voltages and currents at the sending end for a time step. The matrix retains the data for the last cycle. The sampling rate is of 1.041 ms, thus the matrix has 16 rows and 6 columns. A 6-point FFT is taken at every time step for each column of the current matrix. The magnitude of the power spectrum of the Fourier Series is fed into NN (section 3.2.2).

3.5.2 Training Process

The magnitude of the power spectrum of FFT are very sensitive to variations in phase and magnitude of the signals. Thus, a fault cannot be properly characterized by those values obtained for only one cycle during the fault period, as was done for CPN.

Just after the inception and at the ending of the fault the input signals have harmonics which are not present during the fault period. Therefore the power spectrum of these signals is different from the power spectrum during the fault period. Although they are different, they are part of the same fault and ART has to be trained to learn them. This gives a faster identification of the beginning of the fault.

The training set for ART contains as the characteristic feature the magnitude of the power spectrum of the signals at the beginning, the end and at the middle of the fault period. We recall that only the latter one is used for training of CPN.

A very important parameter for ART is the *vigilance factor*. This factor measures the degree of mismatch between the presented pattern and the existing one. It is computed for every pattern presented and is compared to a desired value. If it is lower than the desired value, the pattern is classified as being part of that category, otherwise a new category is created.

As can be envisioned it is very difficult to estimate the optimal value of the Vigilance Factor (VF). A high VF value will create many new classes, and thus ART will loose the ability to sense the common feature of a class. Every small variation in input is seen as a sufficient condition to create a different category. While a lower VF value makes ART to place all the patterns in too few classes. A large variation in input pattern will make no difference and the patterns which need to be in different categories are classified together.

By trial and error, a VF of 0.983 was selected. Since the vigilance factor is high, there are 2-3 classes for each type of fault. Additional post-processing is required to cluster these classes into a single fault class type.

The recall process is done with different data. The same types of fault are applied,

but different fault resistances are used.

3.5.3 Discussion of the Results

The recall phase starts after the training of the NN. The results for three fault cases AG fault, AB fault and ABG fault at the remote end, are presented in Fig. 3.21. These fault cases are similar to those discussed in section 3.3.2.

The AG fault (Fig. 3.21 a) is very well identified, and there is no delay in detection of the inception and end of the fault. A faster detection of the fault inception is possible (by including in the characteristic feature of a fault samples from both transient periods of the fault as part of the fault). During the recall the VF is chosen high enough to obtain the best recall result. A vigilance factor of 0.9 is used during the recall.

Also, for AB fault there is a fast, and correct identification of the beginning of fault and its type. The result is better than that one obtained using CPN (section 3.3). Although, there are instances where the NN indicates a BC fault, the response of the NN is improved. The vigilance factor is 0.95 for this case.

For an ABG fault, there are few instances with false indication. As in the method using CPN (section 3.3), there are oscillations between AB fault, and ABG fault, and

additional instances indicating a BG fault and a BCG fault. The latter ones appears in the transient period at the end of the fault.

In conclusion, ART2 provides a faster identification of the beginning of the fault than CPN. Although there are few instances when the NN does not indicate the proper type of fault, the overall response of the NN is good.

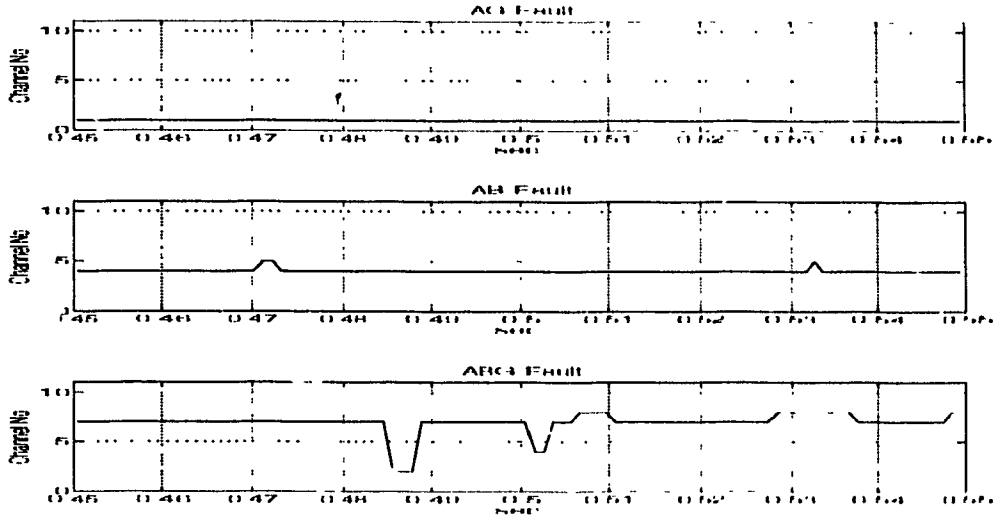

Figure 3.21: ART2 output for fault identification .

Chapter 4

Conclusions & Further Work

4.1 Conclusions

The thesis proposes a novel approach for the protection of series compensated transmission lines. Two different NN architectures were studied, and both architectures offered suitable methods to identify and locate the fault.

One major advantage of the proposed technique over existing traditional methods consists in its ability to function without a telecommunication channel. This means considerable savings in capital and operating costs for the protection system. Furthermore, in the case of traditional methods the lack of a telecommunication link results in improved reliability.

Another major advantage is the adaptability of the NN to a new environment. If the power system parameters or configuration change, a new training set is all that is needed to re-use the protection system based on the NN. All the existing protection equipment remains in place. Using the traditional methods a change in the system parameters/configuration will probably result in a change and re-design of the protection equipment.

The importance of a judicious choice of a training set has been shown in section 3.3. The task of choosing an optimal training set is not trivial. The success of the recall process relies on the ability of the training set to reflect the characteristic features of each specific category. It is also required that the categories be separable.

As has been explained in section 3.3, a small variation in the method of computing the training set can change the results drastically. Using average instead of maximum values for the training set makes a significant difference in the accuracy of the output results.

CPN provides good results for both tasks of identification and location of the fault. One possible limitation of using the CPN is that a new training set will be required if the parameters of the system change. A minor problem with the CPN is a delay of one cycle due to the computation of the RMS values for the input voltage/current signals.

To overcome some of the problems of the CPN a new type of NN architecture (ART2) was used. ART2 can preserve previously acquired experience while learning new categories. Since the method of obtaining the training set is changed, for ART2, the training incorporates the behaviour of the system during the transient period at the beginning and end of the fault. This new information is in addition to that already existing about each type of fault.

Although there were some false indications present during the fault, ART2 provided a faster identification of the beginning of the fault when compared to CPN.

In summary, the novel methods described in this thesis demonstrate the feasibility of using NNs for fault identification and location in power transmission systems.

These methods show enough promise to merit further in depth investigations.

4.2 Further Work

The work described in this thesis has demonstrated the feasibility of using NNs in the identification and location of typical faults in a series compensated transmission line. Much further work is needed before a practical implementation of these techniques can be applied. Some additional work which could not be finished during this project is discussed below:

1. Improvements to the simplified power system simulation model used here are needed to provide a more realistic simulation.

2. The NN architecture used were able to identify only four locations. These NN architectures could be enlarged to provide a finer location of the fault.

3. The NN architecture to locate the fault could be implemented using Back-Propagation (BP) type of NN. The BP is a type of NN that can be trained to approximate functions. The fault location is a function of the line impedance which is given by the relation between the bus voltages and line currents. Thus, a training set can be created to correlate the fault location and the value of the bus voltages and line currents.

Appendix A

Power System Data

Subsystem no. 1 - Sending End

- R = 1.1 Ω, L = 0.1171 H
- shunt: R = 19.49 Ω, L = 5.17 H
- E_m = 624.62 kV, V_0 = 441.67 kV, I_0 = 0.6788 kA

Subsystem no. 2 - Receiving End

- R = 466.2 Ω, L = 0.621 H
- shunt: R = 19.49 Ω, L = 5.17 H

Subsystem no. 3 - Transmission Line

- series capacitor (in the middle of the line): 170 μF
- line length: 214 miles
- line model: 4 Π sections with the following structure

- R = 1.24655 Ω, L = 0.09286 H

- capacitors between phases: 0.1728 μF

- capacitors between line and ground: 0.9164 μF

Appendix B

EMTDC data file

```
0.00005  0.7  0.001 / DELT FINTIM PRTSPT
1                /NUMBER OF SUBSYSTEMS
27 /NUMBER OF NODES IN SUBSYSTEM # 1
0               /NO INITIAL NODE VOLTAGES
1 7 1.1         /A PHASE SENDING END
7 4 0 0.1171    /
-4 0 19.49 5.17      /SHUNT
2 8 1.1         /B PHASE
8 5 0 0.1171    /
-5 0 19.49 5.17      /SHUNT
3 9 1.1         /C PHASE
9 6 0 0.1171    /
-6 0 19.49 5.17      /SHUNT
4 5 100000.0         /to simulate faults at the sending end
```

```
5 6 100000.0          /first 3 are LL

6 4 100000.0          /

4 0 100000.0         /next 3 are LG

5 0 100000.0         /

6 0 100000.0         /

4 25 1.24655 / 1st half of the transmission line

25 10 0 0.092865728  /

5 26 1.24655 /

26 11 0 0.092865728 /

6 27 1.24655 /

27 12 0 0.092865728 /

10 11 0 0 0.172805 / C between phases - these will also be used for faults

11 12 0 0 0.172805 /

12 10 0 0 0.172805 /

10 23 0 0 0.916455 / C between line & ground - see above

11 23 0 0 0.916455 /

12 23 0 0 0.916455 /

10 13 1.0 0. 0. 1. / arrestor

11 14 1.0 0. 0. 1. /

12 15 1.0 0. 0. 1. /

-10 13 0 0 170.0 / series C

-11 14 0 0 170.0 /
```

```
-12 15 0 0 170.0 /
13 14 100000.0              /fault simulation after C LL
14 15 100000.0              /
15 13 100000.0              /
13 23 100000.0              / next 3 are LG
14 23 100000.0              /
15 23 100000.0              /
-13 16 1.24655 0.092865728 / 2nd half of the transmission line
-14 17 1.24655 0.092865728 /
-15 18 1.24655 0.092865728 /
16 17 0 0 0.172805 / C between phases
17 18 0 0 0.172805 /
18 16 0 0 0.172805 /
16 22 0 0 0.916455 / C between line & ground
17 22 0 0 0.916455 /
18 22 0 0 0.916455 /
-16 22 19.49 5.17       /SHUNT
-17 22 19.49 5.17       /
-18 22 19.49 5.17       /
16 19 468.2 /A PHASE RECEIVING END
19 22 0 0.621 /
17 20 468.2 /B PHASE RECEIVING END
```

```
20 22 0 0.621 /
18 21 468.2 /C PHASE RECEIVING END
21 22 0 0.621 /
-22 23 9.1271 0.076044686 / ground R-L
23 24 0 0.076044686 /
24 0 9.1271 /
999                 /END OF BRANCH DATA FOR SUBSYSTEM # 1
1 0 0.00004         /A PHASE SENDING END SOURCE
2 0 0.00004         /B PHASE
3 0 0.00004         /C PHASE
999 /               END OF SOURCE DATA FOR SUBSYSTEM # 1
999 /               END OF MUTUAL DATA FOR SUBSYSTEM # 1
999                 /END OF TRANSMISSION LINE DATA
-1000 +1000         /SCREEN PLOT LIMITS
20                  /NUMBER OF PGB'S
624.62  0.45  0.1  0.0  /VARS A phase voltage magnitudes,START FAULT, LENGTH
```

Appendix C

EMTDC subroutine to control the dynamics of the power system

```
      SUBROUTINE DSDYN
C
C   *** FORTRAN 77 ***          date 10/10/91
C
C   Associated files -
C      Datafile: DATATB1
C         Dsout: DSOUTTB1.FOR
C
C   Purpose - Transmission line (332 mi)
C
C
C   Include and Common Block declarations
C   -------------------------------------
```

```
      INCLUDE 'emt.e'
      COMMON /S1/TIME,DELT,ICH,PRINT,FINTIM
      COMMON /S2/STOR(10000),NEXC
      COMMON /S3/GVLV(4,4,40),NVLV
      COMMON /S4/VAR(200),CON(200),PGB(75)
C
C   Variable definitions
C   --------------------
C   INTGL3: CSMF integral function with limits
C   LDLAG2: CSMF lead/lag function
C    LIMIT: CSMF limiter function
C
C   Variable declarations
C   ---------------------
      REAL INTGL3,LDLAG2,LIMIT
C
C***********************************************
C  LOCAL VARIABLE TO SIMULATE PHASE TO PHASE FAULTS
C  WITHOUT DECLARE THEM IN DATA FILE
C
      LOGICAL AFTERFLT
      REAL G1,G2,G3
```

```
C************************************************
C   CON(1..100) definitions
C   -----------------------
C    ***
C
C   VAR(1..100) definitions
C   -----------------------
C    VAR(1)=A phase step voltage magnitude
C    VAR(2)=START TIME FOR THE FAULT
C    VAR(3)=LENGTH OF THE FAULT
C    VAR(4)=TYPE OF FAULT:
C           - 1=AG, 2=BG, 3=CG, 4=AB, 5=BC
C           - 6=CA, 7=ABG, 8=BCG, 9=CAG, 10=ABCG
C
C
C   Initialization steps
C   --------------------
      IF (TIME .GT. DELT) GO TO 10
        AFTERFLT=.FALSE.
  10  CONTINUE
C
C   Program begins
```

```
C    -------------
C
C
C
C    NO FAULT INITIALIZATION
C
         IF (AFTERFLT) THEN
           GDC(16,17,1)=G1
           GDC(17,16,1)=G1
           GDC(17,18,1)=G2
           GDC(18,17,1)=G2
           GDC(18,16,1)=G3
           GDC(16,18,1)=G3
           GDC(16,22,1)=G4
           GDC(22,16,1)=G4
           GDC(17,22,1)=G5
           GDC(22,17,1)=G5
           GDC(18,22,1)=G6
           GDC(22,18,1)=G6
           AFTERFLT=.FALSE.
         ENDIF
C
```

```
C   START SECTION
C
        VRAMP=VAR(1)
        IF (TIME.LT.0.1) VRAMP=VAR(1)*(TIME*11.111111-0.111111)
        IF (TIME.LT.0.01) VRAMP=0.0
C
C   FAULT SECTION
C
        IF ((TIME.GT.VAR(2)).AND.(TIME.LT.(VAR(2)+VAR(3))).AND.(VAR(
     +4).NE.0.0)) THEN
          G1=GDC(16,17,1)
          G2=GDC(17,18,1)
          G3=GDC(18,16,1)
          G4=GDC(16,22,1)
          G5=GDC(17,22,1)
          G6=GDC(18,22,1)
          IF ((VAR(4).EQ.1).OR.(VAR(4).EQ.7).OR.(VAR(4).EQ.9)
     +.OR.(VAR(4).EQ.10)) GDC(16,22,1)=10.0
          IF ((VAR(4).EQ.2).OR.(VAR(4).EQ.8).OR.(VAR(4).EQ.7)
     +.OR.(VAR(4).EQ.10)) GDC(17,22,1)=10.0
          IF ((VAR(4).EQ.3).OR.(VAR(4).EQ.9).OR.(VAR(4).EQ.8)
     +.OR.(VAR(4).EQ.10)) GDC(18,22,1)=10.0
```

```
          IF (VAR(4).EQ.4) THEN
            GDC(16,17,1)=10.0
            GDC(17,16,1)=10.0
          ENDIF
          IF (VAR(4).EQ.5) THEN
            GDC(17,18,1)=10.0
            GDC(18,17,1)=10.0
          ENDIF
          IF (VAR(4).EQ.6) THEN
            GDC(18,16,1)=10.0
            GDC(16,18,1)=10.0
          ENDIF
          GDC(22,16,1)=GDC(16,22,1)
          GDC(22,17,1)=GDC(17,22,1)
          GDC(22,18,1)=GDC(18,22,1)
          AFTERFLT=.TRUE.
        ENDIF
C
        CALL ESYS1(1,1,2,3,VRAMP,377.0,0.0)
C
      RETURN
      END
```

Appendix D

EMTDC output subroutine

```
      SUBROUTINE DSOUT
C
C   *** FORTRAN 77 ***          date 10/10/91
C
C   Associated files -
C      Datafile: DATATB1
C          Dsdyn: DSDYNTB1.FOR
C
C   Purpose - Transmission line
C
C
C   Include and Common Block declarations
C   -------------------------------------
      INCLUDE 'emt.e'
```

```
      COMMON /S1/TIME,DELT,ICH,PRINT,FINTIM
      COMMON /S2/STOR(10000),NEXC
      COMMON /S3/GVLV(4,4,40),NVLV
      COMMON /S4/VAR(200),CON(200),PGB(75)
C
C   Variable definitions
C   --------------------
C   INTGL3: CSMF integral function with limits
C   LDLAG2: CSMF lead/lag function
C    LIMIT: CSMF limiter function
C
C   Variable declarations
C   ---------------------
      REAL INTGL3,LDLAG2,LIMIT
C
C   CON(1..100) definitions
C  .-----------------------
C    ***
C
C   VAR(1..100) definitions
C   -----------------------
C    ***
```

```
C
C   PGB(1..25) definitions
C   ----------------------
C     PGB(1)=A phase sending end voltage
C     PGB(2)=B            "
C     PGB(3)=C            "
C     PGB(4)=A phase receiving end voltage
C     PGB(5)=B            "
C     PGB(6)=C            "
C     PGB(7)=A phase sending end line current
C     PGB(8)=B            "
C     PGB(9)=C            "
C     PGB(10)=A phase receiving end line current
C     PGB(11)=B               "
C     PGB(12)=C               "
C     PGB(13)=REAL POWER FOR SEND END
C     PGB(14)=REACTIVE POWER FOR SEND END
C     PGB(15)=REAL POWER FOR RECV END
C     PGB(16)=REACTIVE POWER FOR RECV END
C   Initialization steps
C   --------------------
      IF (TIME .GT. DELT) GO TO 10
```

```
C      ***
  10  CONTINUE
C
C   Program begins
C   --------------
C
      PGB(1)=VDC(4,1)
      PGB(2)=VDC(5,1)
      PGB(3)=VDC(6,1)
C
      PGB(4)=VDC(16,1)-VDC(22,1)
      PGB(5)=VDC(17,1)-VDC(22,1)
      PGB(6)=VDC(18,1)-VDC(22,1)
C
      PGB(7)=CDC(4,7,1)
      PGB(8)=CDC(5,8,1)
      PGB(9)=CDC(6,9,1)
C
      PGB(10)=CDC(16,19,1)
      PGB(11)=CDC(17,20,1)
      PGB(12)=CDC(18,21,1)
C
```

```
      PGB(13)=P3PH2(1,4,5,6,7,8,9,0.001)
      PGB(14)=Q3PH2(1,4,5,6,7,8,9,0.001)
      PGB(15)=-P3PH2(1,16,17,18,19,20,21,0.001)
      PGB(16)=-Q3PH2(1,16,17,18,19,20,21,0.001)
C
      PGB(17)=VDC(10,1)-VDC(13,1)
      PGB(18)=VDC(11,1)-VDC(14,1)
      PGB(19)=VDC(12,1)-VDC(15,1)
C
      PGB(20)=CDC(24,24,1)
      RETURN
      END
```

Appendix E

Neural Networks

CPN Structure for Fault Identification

- input neurons: 6
- hidden neurons: 12
- output neurons: 11

CPN Structure for Fault Identification and Location

- input neurons: 6
- hidden neurons: 85
- output neurons: 15

ART2 Structure for Fault Identification

- input neurons: 36
- output neurons (max.): 48

Bibliography

[1] Phadke, A. G., J. S. Thorp, *Computer Relaying for Power Systems*, John Wiley & Sons Inc., 1988.

[2] Gönen, T., *Modern Power System Analysis*, John Wiley & Sons Inc., 1988.

[3] Gönen, T., *Electric Power Distribution System Engineering*, McGraw-Hill Book Comp., 1986.

[4] Thomas, D.W.P., C. Christopoulos, *"Ultra-high Speed Protection of Series Compensated Lines"*, IEEE/PES 1991, Summer Meeting, San Diego, California, July 28 - August 1, 1991, Paper No. 91-SM-359-0-PWRD.

[5] Larsen, E.V., Luc Gerin-Lajoie, *"Basic Aspects of Applying SVC's to Series-Compensated AC Transmission Lines"*, IEEE/PES 1990 Winter Meeting, Atlanta, Georgia, February 4 - 8, 1990, Paper No. 90-WM-080-2-PWRD.

[6] Christopoulos, C., D.W.P. Thomas, Arthur Wright, *"Scheme based on travelling-waves, for the protection of major transmission lines"*, IEE Proceedings, Vol.

135, Pt. C, No. 1, pp. 63 - 73, January 1988.

[7] Khaparde, S.A., P.B. Kale, S.H. Agarwal *"Application of Artificial Neural Networks in Protective Relaying of Transmission Lines"*, Procedure of First International Forum on Applications of Neural Networks to Power Systems, Edited by M.El-Sharkawi and R. Marks, July 23-26, 1991, Seattle, Washington.

[8] Champagne L., J.P. Benoit, *"Protection of Hydro-Quebec's Series-Compensated Lines"*, Canadian Electrical Association, May 1991, Toronto.

[9] Newbould, A., P. Hindle, *"Series Compensated Lines: Application of Distance Protection"*, 14^{th} Annual Western Protective Relay Conference, Spokane, Washington, Oct. 1987.

[10] Aggoune, M., M.A. El-Sharkawi, D.C. Park, M.J. Damborg and R.J. Marks II, *"Preliminary Results on Using Artificial Neural Networks for Security Assessment"*, PICA 1989, Seattle, Washington, May 1 - 5, 1989, pp. 252 - 258, Paper No. CH2747-4/89/0000-0252.

[11] Indulker, C.S., B. Viswanathan, S.S. Venkata, *"Maximum power Transfer Limited by Voltage Stability in Series and Shunt Compensated Schemes for AC Transmission Systems"*, IEEE Transactions on Power Delivery, Vol. 4, No. 2, pp. 1246 - 1252.

[12] Cardozo, E., S.N. Talukdar, "*A Distributed Expert System for Fault Diagnosis*", PICA 1987, Montreal, Canada, May 18 - 21, 1987, pp. 611 - 616, Paper No. 0885-8950/88/0500-0641.

[13] Girgis, A.A., M.B. Johns, "*A Hybrid Expert System for Fault Section Identification, Fault Type Classification and Selection of Fault Location Algorithms*", IEEE Transactions on Power Delivery, Vol. 4, No. 2, April 1989, pp. 978 - 985.

[14] Sobajic, D.J., Pao Yoh-Han, "*An Artificial Intelligence System for Power System Contingency Screening*", PICA, Montreal, Canada, May 18-21, 1987, Transactions on Power Systems, Vol. 3, No.2, May 1988, pp. 647-653.

[15] Okada, K., Urasawa, K., Kanemaru, K., Kanoh, H., "*Knowledge-based Fault Location System for Electric Power Transmission Lines with OPGW*", International Workshop on Artificial Intelligence for Industrial Applications 1988, pp. 52-57.

[16] Wasserman, P.D., *Neural Computing - Theory and Practice*, Van Nostrand Reinhold, New York, 1989.

[17] Sachdev, M.S., L.R. Tumma, "*Protection of Series Compensated Transmission Lines - Modelling and Appraising*", CEA Spring Conference, 1991.

[18] Kandil, N., V.K. Sood, K.Khorasani, R.V. Patel, "*Fault Identification in an AC Transmission System using Neural Networks*", IEEE Trans on Power Systems, May 1992, Vol. 7, No. 2, pp 812-819.

[19] *The EMTDC User's Manual*, Manitoba HVDC Research Center, Winnipeg, Manitoba, Canada.

[20] *The NeuralWare Inc.*, Neuralworks, rev 2.00 Neuralware Inc., Sevickley, PA. 1988.