ELSEVIER

Electric Power Systems Research 34 (1995) 109–116

ELECTRIC POWER SYSTEMS RESEARCH

High-speed fault detection and classification with neural nets

M. Kezunović [a], Igor Rikalo [a], D.J. Šobajić [b]

[a] *Department of Electrical Engineering, Texas A&M University, College Station, TX 77843-3128, USA*
[b] *Electric Power Research Institute, 3412 Hillview Avenue, Palo Alto, CA 94304-1395, USA*

Received 7 February 1995

Abstract

This paper introduces a new neural net (NN) approach for automated fault disturbance detection and classification. The NN design and implementation are aimed at high-speed processing which can provide selective real-time detection and classification of faults. The approach is extensively tested using the Electromagnetic Transients Program (EMTP) simulations of two quite complex transmission system configurations. The results indicate that the speed and selectivity of the approach are quite adequate for a number of different transmission and distribution monitoring, control, and protection applications.

Keywords: Neural networks; Data processing; Fault analysis; Electromagnetic Transients Program (EMTP) simulations

1. Introduction

Automated analysis of power system faults has recently become an application of wide interest in both transmission and distribution areas. This interest has been generated since the automated analysis can be utilized in a number of functions such as alarm processing, fault diagnosis, and restorative switching. Several systems supporting this application have been implemented in the past using information obtained from circuit breaker and relay contacts. The information was acquired by either the sequence of events recorders (SOEs) or the remote terminal units (RTUs) of the supervisory control and data acquisition (SCADA) systems. A number of such developments have been reported worldwide [1].

Most recently, it has been recognized that the fault analysis can be enhanced significantly by considering samples of the analog quantities (voltages and currents) in addition to the contacts. The authors of this paper have developed a new concept for automated fault analysis based on samples of both analog and contact information acquired by the digital fault recorders (DFRs) [2]. Based on this concept, a system was designed and implemented for Houston Lighting and Power (HL&P) Company [3].

The HL&P system was aimed at automating fault disturbance analysis which would help the operators significantly by relieving them from the tedious, and quite often time consuming, task of manually searching and analyzing a large number of DFR records. After an extensive series of tests using actual data from the field, it has been confirmed that the system provides extremely fast and completely automated operation. As a result, the system has recently been installed at the South Texas Project (STP) switchyard for further use by HL&P Engineering Design, Operations and Maintenance Departments [4].

This paper reports on further enhancements of the automated fault analysis that are achieved using neural net (NN) processing. Based on the experiences from the HL&P system development, it has been recognized that the computation of signal parameters and the logic processing used to compare signal parameters with predefined settings can be substituted by a neural network. Similar approaches were reported by some other authors studying problems of waveform estimation [5] and classification [6], as well as fault direction discrimination [7] and fault classification [8].

The goal of the study reported in this paper was the implementation of a new NN approach that would demonstrate some improvements over the existing NN techniques. A new processing method that incorporates the advantages of both supervised and unsupervised training procedures is developed. Preliminary results were quite encouraging [9,10]. In order to fully demonstrate the potential benefits of the new NN approach, a high-speed fault disturbance detection and classification scheme was designed, implemented, and tested. This

SSDI 0378-7796(95)00962-H

paper gives details of this study with elaborate evaluation of results obtained through computer simulations of faults using two complex Electromagnetic Transients Program (EMTP) models of actual power system sections.

The first part of the paper is devoted to a discussion of the application framework for this novel high-speed technique. A new method of supervised clustering is presented for the transmission line fault identification problem. A detailed description of the NN design and implementation is given next. A summary of the evaluation setup and results obtained using EMTP simulations are given at the end.

2. Application framework

For a clear understanding of the benefit achieved with the new NN approach, it is important to emphasize the specific properties and characteristics of both the power system application and the neural net implementation.

The fault analysis application considered in this paper is related to fault detection and fault type classification, and embodies several data processing properties [11]. Data acquisition is aimed at collecting samples of analog quantities (voltage and currents) from the secondaries of instrument transformers, and status information (contacts) from circuit breakers, switches, and relays. The samples of analog quantities need to be processed simultaneously for all the voltages and currents on a transmission line. This facilitates timely determination and comparison of the signal parameters and time sequences of contact changes. The process of comparison requires easy interfacing between the signal and logic processing. The final outcome of the fault analysis can be obtained with high selectivity and speed since all the decisions made are based on instantaneous changes of the signal parameters and the corresponding sequence of events.

The fault analysis application, as defined in the context of this paper, requires that fault detection and classification are determined in the following manner.

- The processing has to be performed in time to allow system operators to use the outcome of the analysis in online applications.
- The outcome of the processing has to be presented in a symbolic form (class names) since the detection and classification results of the neural net computation may be further utilized in a rule based expert system.
- The NN training has to be quite efficient and straightforward since the fault analysis application requires a fast and simple procedure to adapt to the changing power network conditions.

Fault detection and classification is defined as a multiclass problem. The eleven types of faults (a–g, b–g, c–g, a–b, b–c, c–a, ab–g, bc–g, ca–g, abc, abc–g) and the no-fault situation produce a 12-class classification problem.

A literature search indicates that most of the NN implementations for fault detection and classification are based on multilayer feedforward nets. In this case the application is considered to be a mapping problem. Supervised learning can be used where sets of associated input–output pairs are presented to a net which then 'learns' a model of that process. However, the training of multilayer networks is computationally demanding and in some instances tens of thousands of iterations are needed to achieve convergence. Such a performance may not be suitable for fast detection and classification. Since our problem is a classification problem, where only discrete labeling of classes is needed, the use of feedforward networks may not be fully justified under stringent processing time requirements.

Another possible approach of the NN application to our problem is to exploit data self-organization obtained through the use of unsupervised learning. After the learning (cognition phase), the user defines or labels clusters according to some criterion. The net is then ready for the classification task (recognition phase). Therefore, the concept of data self-organization through the use of unsupervised learning is valuable for discovering how an ensemble of patterns is distributed in the pattern space.

To overcome the above-mentioned limitations of multilayer feedforward networks, and to take advantage of the suitability of self-organizing networks to perform a classification through the clustering process, a new NN approach has been developed and applied in our study. It incorporates the advantages of both supervised and unsupervised training procedures and yet meets the requirements presented earlier. The proposed method utilizes the concept of supervised clustering which demonstrates the following important properties.

- The number of iterations in the learning process is greatly reduced using unsupervised learning with a supervised class membership inheritance process.
- The training is far less complex than in standard supervised learning.
- Combination of symbolic and numeric data is readily available.

3. NN description

The NN algorithm used for this study embodies the ISODATA clustering algorithm which is well known in classical pattern recognition [12]. This type of neural net assumes no teaching and performs unsupervised learning. The process performs a comparison of a given input with previously encountered patterns. If the input is similar to any of the patterns, it will be placed in the same category. If the input is not similar to any of the previously presented patterns, a new category will be assigned. Category proliferation is controlled by the threshold parameter. A NN system with a low threshold will permit grouping of patterns with high similarity and vice versa.

Fig. 1 shows the block diagram of the algorithm that combines both unsupervised learning (USL) and supervised learning (SL), used in this study. The initial data set, containing all the patterns, is processed using an unsupervised clustering algorithm.

Initially, the threshold parameter ρ is large and chosen in such a way that the clustering algorithm generates a small number of clusters. As the training process continues, the value of ρ is reduced (e.g. after every interation $\rho_{\text{new}} = 0.95\rho_{\text{old}}$).

The output is a stable family of clusters, defined as hyperspheres in N-dimensional space, where N denotes the number of features in each pattern. The task of supervised learning is to separate nonhomogeneous clusters from the homogeneous ones.

Next, class membership is assigned to homogeneous clusters. The training data set is then reduced to contain only patterns from nonhomogeneous clusters. The threshold parameter ρ is decreased, and the whole procedure is reiterated.

After completion of the training procedure, all generated clusters contain uniform data patterns, and are characterized by their centroids, corresponding radii (i.e. threshold parameter ρ), and inherited class membership. Fig. 2 shows a schematic illustration of the outcome of the training process in the feature space.

It can be observed that the cluster topology is not uniform, and that two or more clusters may have the same class membership.

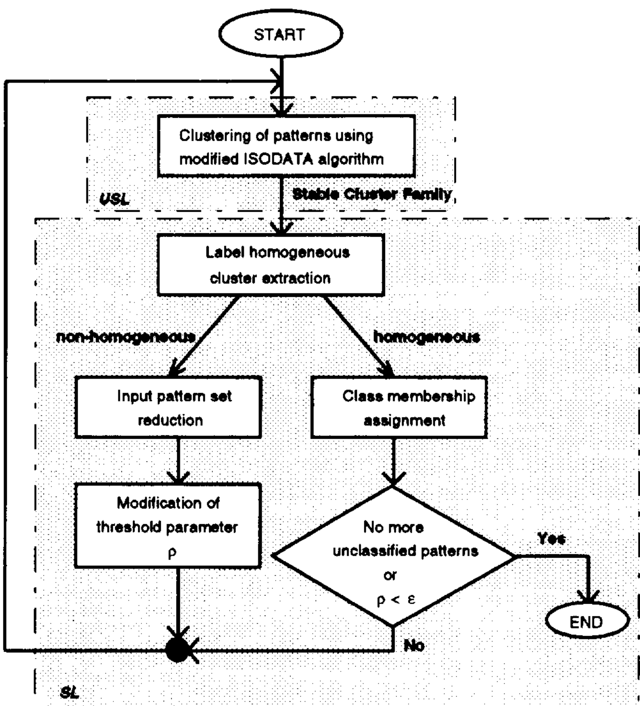

Fig. 1. Artificial neural network learning process.

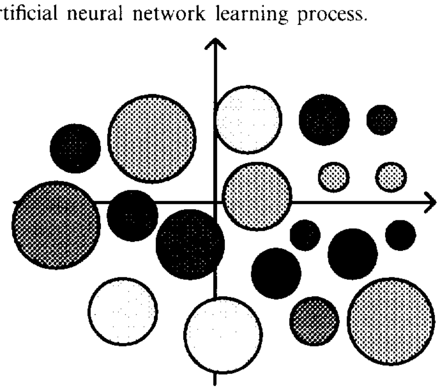

Fig. 2. Schematic illustration of the outcome of the training process.

4. Implementation details

The mathematical foundation of the NN used is described as follows.

Given is a set of P ($p = 1, 2, \ldots, P$) patterns, $\boldsymbol{x}^{(p)}$, where

$$\boldsymbol{x}^{(p)} = [x_1^{(p)}, x_2^{(p)}, \ldots, x_N^{(p)}]^{\mathrm{T}} \tag{1}$$

4.1. Initialization run

Step 1. We form cluster number 1, $\boldsymbol{b}_1(1) = \boldsymbol{x}^{(1)}$ (meaning cluster C_1 with centroid $\boldsymbol{b}_1$ contains one pattern).

Step 2. If $(\boldsymbol{x}^{(2)} - \boldsymbol{b}_1)^{\mathrm{T}}(\boldsymbol{x}^{(2)} - \boldsymbol{b}_1) \leqslant \rho^2$, then we adapt $\boldsymbol{b}_1$ as

$$\boldsymbol{b}_1(2) = \boldsymbol{b}_1(1) + \tfrac{1}{2}[\boldsymbol{x}^{(2)} - \boldsymbol{b}_1(1)] \tag{2}$$

If $(\boldsymbol{x}^{(2)} - \boldsymbol{b}_1)^{\mathrm{T}}(\boldsymbol{x}^{(2)} - \boldsymbol{b}_1) > \rho^2$ then we form cluster 2 as

$$\boldsymbol{b}_2(1) = \boldsymbol{x}^{(2)}$$

In doing so, after presenting $q < P$ patterns the situation is as follows: m clusters exist, their centroids $\boldsymbol{b}_m$ are known and we know how many patterns, n_m, belong to each cluster.

When we present the next pattern, $q + 1$, we first allocate the closest cluster τ by

$$\min_j [(\boldsymbol{x}^{(q+1)} - \boldsymbol{b}_j)^{\mathrm{T}}(\boldsymbol{x}^{(q+1)} - \boldsymbol{b}_j)] = r_\tau^2 \tag{3}$$

and then compare r_τ^2 and ρ^2.

If $r_\tau^2 \leqslant \rho^2$ then we adapt the cluster as

$$\boldsymbol{b}_\tau(n_\tau + 1) = \boldsymbol{b}_\tau(n_\tau) + \frac{1}{n_\tau + 1}[\boldsymbol{x}^{(q+1)} - \boldsymbol{b}_\tau(n_\tau)] \tag{4}$$

If $r_\tau^2 > \rho^2$ then we form a new cluster as

$$\boldsymbol{b}_{m+1}(1) = \boldsymbol{x}^{(q+1)}$$

This procedure is repeated until the entire set of patterns is processed once.

4.2. Stabilization run

Step 3. We present every pattern, $x^{(p)}$, again. Let, say, the present pattern p belong to cluster C_k. The shortest distance between $x^{(p)}$ and all existing centroids b_j is found using Eq. (3).

If $\tau = k$ and $r_\tau^2 \leqslant \rho^2$ then no learning occurs; we check the next pattern, $p+1$.

If $\tau \neq k$ and $r_\tau^2 \leqslant \rho^2$ then we adapt b_τ using Eq. (4) and b_k as

$$b_k(m_k - 1) = b_k(n_k) - \frac{1}{n_k - 1}[x^{(p)} - b_k(n_k)] \tag{5}$$

for $n_k > 1$.

If $r_\tau^2 > \rho^2$ we form the new cluster C_m,

$$b_m(1) = x^{(p)}$$

and adapt the 'previous' centroid b_k using Eq. (5).

Stabilization is repeated until no patterns change their cluster membership.

5. Test system configurations

The Electromagnetic Transients Program (EMTP) was chosen as a simulation tool for providing training data [13]. EMTP was chosen because it is widely used software for accurate and detailed simulation of electromagnetic, electromechanical, and control system transients on multiphase electric power systems.

Two different power systems were modeled using EMTP. Simulations, including all types of transmission line faults, as well as different fault locations and fault resistances were conducted. More than 3000 different fault patterns were generated for both models. Since EMTP simulations are computationally intensive, an IBM RISC 6000/340 workstation was used. Supporting software and tools to facilitate and automate the simulation process have also been developed. The EMTP simulations took approximately 30 hours of computer time.

System 1 represents a section of an actual 161 kV power system with short and mutually coupled transmission lines. A single-line diagram of this system is given in Fig. 3. The transmission line along which the fault events were simulated is fully transposed and 21.5 km long (line between buses 2 and 3). Details of the EMTP model are given in the Appendix.

System 2 is more complex. It represents a section of an actual 345 kV power system with long and compensated transmission lines. EMTP representation also contains detailed models of capacitor coupled voltage

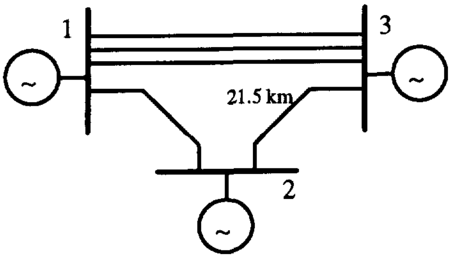

Fig. 3. Single-line diagram of the 161 kV power system.

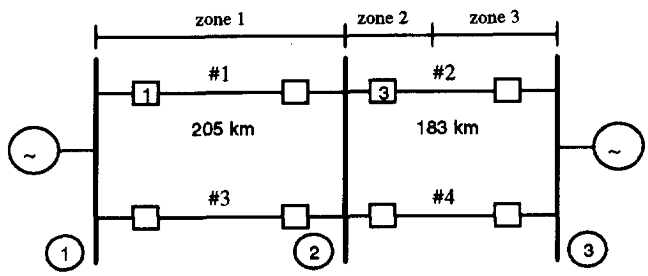

Fig. 4. Single-line diagram of the 345 kV power system.

transformers (CCVTs), current transformers (CTs), metal oxide varistors (MOVs), and surge arresters. For clarity, these elements are not all shown in the single-line diagram in Fig. 4. Part of lines 1 and 3 and lines 2 and 4 is mutually coupled. Faults were simulated on lines 1 and 2. Protection zones indicated in Fig. 4 are selected based on the ideal relay operating characteristic. Relay 1 should operate instantaneously for the faults in zone 1. In the case of faults in zones 2 and 3, relay 1 should operate after T_2 and T_3 seconds ($T_2 < T_3$), thus allowing relay 3 to operate first and clear the fault. This relay operating logic is emulated in the NN algorithm described in this paper. The model details are presented in the Appendix.

6. Test results

In order to evaluate the performance of our NN approach, several different test strategies were tried.

First, the NN was subjected to different types of input signals. Table 1 shows six input data sets that were used for NN training and testing. The concept of

Table 1
Neural network input data sets

Input format	Pattern length		Sample types
	Cycles	No. of samples	
1	3	594	voltages and currents
2	3	594	only currents
3	3	297	only voltages
4	1	198	voltages and currents
5	1	99	only currents
6	1	99	only voltages

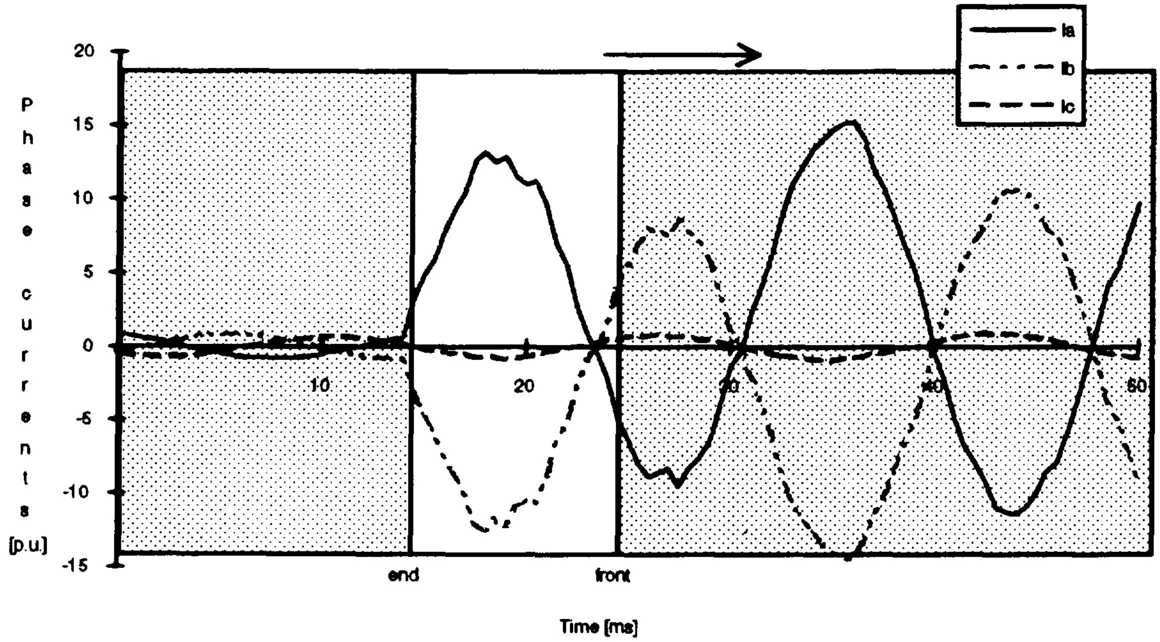

Fig. 5. Sliding window NN input for input set 2.

a sliding window input into the NN for one of the input types is shown in Fig. 5. Sliding motion is obtained by putting every new sample at the front of the window and removing the first sample from the end of the window. This arrangement was used for evaluation of real-time applications.

Second, two different NN training procedures were tried, one with a reduced number of training samples (case 1), and the other using a large number of training samples (case 2). Table 2 shows a summary of these two cases used for evaluation of non-real-time applications.

For real-time applications, the NN classifier has to perform fault detection and fault type classification, typically in one 60 Hz cycle (16.67 ms). The NN classifier was trained using 200 fault patterns. For this application, fault cases were generated using the 161 kV model. The input feature vector to the NN contained samples of phase currents (all three phases). It was organized in the form of a sliding data window with a fixed window length of one cycle (33 samples at 2 kHz). The NN was implemented and tested on an IBM compatible PC with an Intel 486 DX-2 microprocessor. The computation time for the fault detection logic was 0.2 ms, and for the fault classification logic it was 15 ms. Taking into consideration that the data sampling frequency was 2 kHz, it can be concluded that the fault detection was operating in real time, while fault classification required slightly longer. Further work is being done to evaluate possible enhancements of the NN computational performance by using special digital signal processors (DSPs) for real-time embedded applications.

After training, the NN was tested using 80 new fault cases that were generated using the 161 kV EMTP model. Test patterns included all types of transmission line faults, different fault resistances, and various fault incidence angles. It has been observed that the fault incidence angle is the parameter that has the most important influence on the overall NN performance. The classification rate that the NN reached for this application ranged from 52% for faults that had incidence angles different from those used for initial training to 92% for new faults that had incidence angles similar to those used for training.

Table 2
Neural network training cases

Case	Phase	No. of Patterns	Type of patterns
1	Training	397	patterns covering only boundaries of zones 1, 2, and 3 all fault resistances low
	Testing	1980	patterns covering all three zones, every 10% of lines, fault resistances low and high
2	Training	1189	patterns covering all three zones, every 10% of lines, fault resistances both low and high
	Testing	1188	patterns covering all three zones, every 10% of lines, fault resistances low and high

The offline applications were tested using more than 2000 different fault cases obtained from the 345 kV EMTP model. Two training cases were evaluated (see Table 2). First, training of the NN was conducted using 397 patterns. These fault patterns included all types of line faults, different fault resistances, and different fault incidence angles, covering only the boundaries of zones 1, 2, and 3. Lines 1 and 2 (see Fig. 4) were segmented and faults were simulated every 10% of the length of both lines. Faults generated on line 2 were regarded as remote faults with respect to line 1. The training procedure for this NN is very fast. Less than 10 minutes of the IBM RISC 6000/340 computer time were needed to complete the training.

The testing of the NN was performed using 1980 new fault cases. The classification rates are shown in Fig. 6.

It can be seen that the NN exhibits good generalization capabilities even when subjected to restricted training (case 1). Classification rates for fault type and fault location are similar.

After a close inspection of misclassified fault patterns, it was noticed that the largest number of misclassified patterns belong to the faults with high resistance. Fig. 7 shows the classification results when these patterns were removed from the test set. The worst classification results were obtained when only voltage

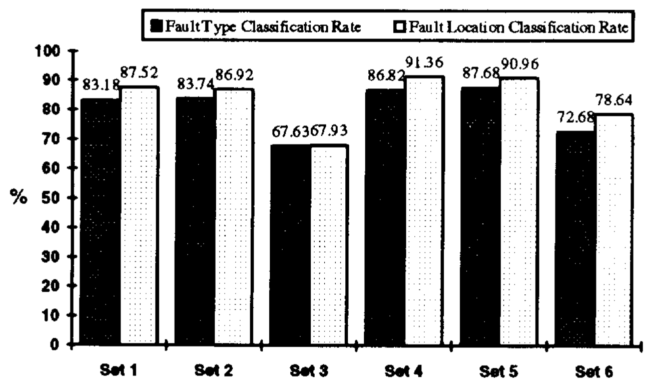

Fig. 6. NN classification results for case 1.

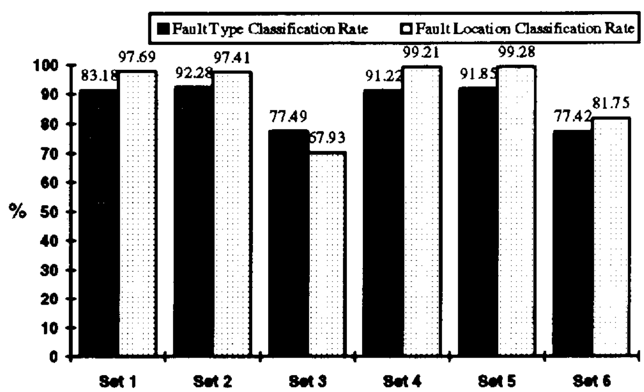

Fig. 7. NN classification results for case 1 without high-resistance faults.

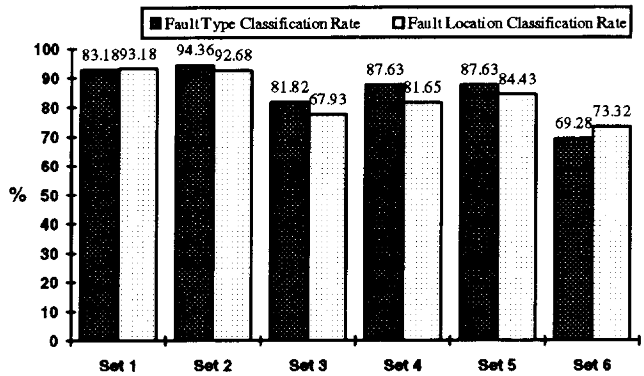

Fig. 8. NN classification results for case 2.

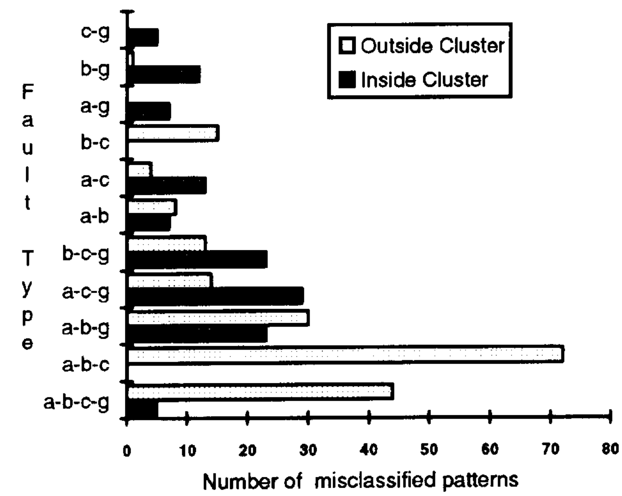

Fig. 9. NN fault type misclassification results for input data set 2.

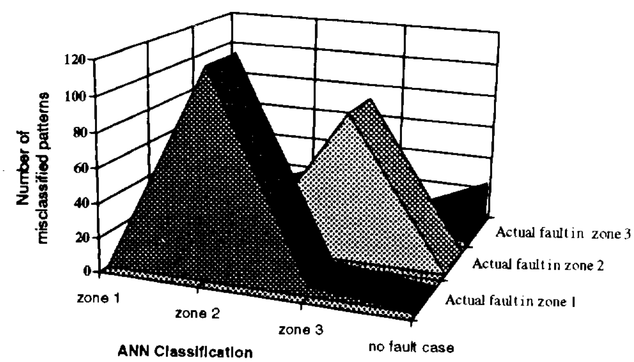

Fig. 10. NN fault location misclassification results for input data set 2.

signals were used as NN inputs.

Training case 2 contains 1189 fault patterns. These patterns included all types of line faults, different fault resistances, and different fault incidence angles, covering all three zones. Faults were simulated as before, every 10% of the length of lines 1 and 2. Faults generated on line 2 were regarded as remote faults with respect to line 1. The training procedure in this case took less than an hour on the IBM RISC 6000/340 workstation.

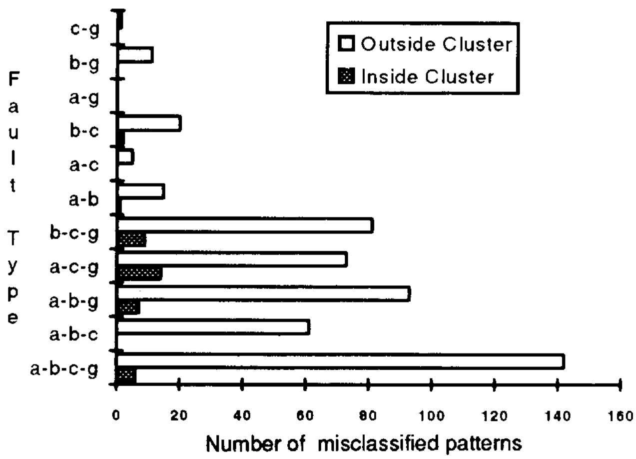

Fig. 11. NN fault type misclassification results for input data set 6.

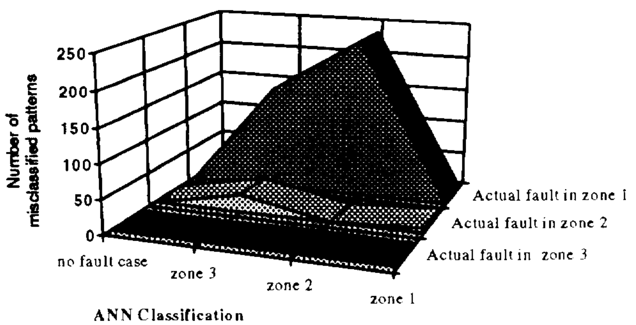

Fig. 12. NN fault location misclassification results for input data set 6.

The testing was then performed using 1188 new fault cases. The NN classification rates are shown in Fig. 8. It can be seen that the classification rate is greater than in case 1. This was expected because of the more appropriate NN training.

Figs. 9–12 present the distribution of the misclassified patterns by fault type (Figs. 9 and 11) and fault location (Figs. 10 and 12). The best results were obtained for input data set 2 (Figs. 9 and 10) where inputs were only phase current waveforms with a data window length of three cycles. On the other hand, the worst performance was noticed for input data set 6 (Figs. 11 and 12) where only voltage waveforms were presented and the data window was one cycle long.

7. Conclusions

The results of this study demonstrate that:

- the proposed NN approach is quite powerful since it combines the advantage of supervised and unsupervised learning techniques;
- stringent application requirements for high speed and selectivity are met;
- classification consistency is maintained over a wide range of testing conditions.

Appendix

Table A1
Source impedances for system 1

Bus		Per-unit value	Actual value (Ω)
1	Z_0	0.58 + j6.32	1.503 + j16.382
	Z_1	0.58 + j11.41	1.503 + j29.576
2	Z_0	0.07 + j1.07	0.181 + j2.774
	Z_1	0.04 + j0.73	1.104 + j1.892
3	Z_0	0.75 + j4.07	1.944 + j10.550
	Z_1	0.31 + j3.04	0.804 + j7.880

Table A2
System equivalents for system 1

Line		Per-unit value	Actual value (Ω)
1–3	Z_0	119.69 + j188.9	310.25 + j489.72
	Z_1	1.80 + j11.44	4.67 + j29.65
2–3	Z_0	∞	∞
	Z_1	12.58 + j74.00	32.61 + j191.82
1–2	Z_0	39.79 + j100.63	103.14 + j260.84
	Z_1	2.75 + j18.32	7.13 + j47.49

Table A3
Self-impedances of lines for system 1

Bus		Per-unit values	Actual value (Ω)
1–3	Z_0	8.94 + j28.34	23.18 + j73.40
	Z_1	1.52 + j9.00	3.94 + j23.48
1–3	Z_0	8.52 + j29.23	22.08 + j75.71
	Z_1	1.38 + j8.80	3.58 + j76.13
1–3	Z_0	8.40 + j29.37	21.77 + j76.13
	Z_1	1.34 + j8.73	3.47 + j22.62
1–2	Z_0	8.42 + j26.74	21.82 + j69.31
	Z_1	1.50 + j8.47	3.88 + j21.95
2–3	Z_0	3.67 + j12.38	9.51 + j32.09
	Z_1	0.67 + j3.92	1.73 + j10.16

Table A4
Lines 1 and 3 (with mutual coupling): system 2

Phase	R' (Ω/km)	Z_c (Ω)	τ (s)	Length (km)
b	7.07E − 01	7.53E + 02	7.10E − 04	129.55
a	5.13E − 02	4.69E + 02	4.94E − 04	129.55
c	3.13E − 02	3.32E + 02	4.45E − 04	129.55
a	3.06E − 02	3.18E + 02	4.40E − 04	129.55
b	3.03E − 02	2.78E + 02	4.37E − 04	129.55
c	3.01E − 02	2.75E + 02	4.37E − 04	129.55

Table A5
Lines 1 and 3 (no mutual coupling): system 2

Phase	R' (Ω/km)	Z_c (Ω)	τ (s)	Length (km)
b	3.63E − 01	6.38E + 02	3.73E − 04	75.31
a	3.03E − 02	2.78E + 02	2.54E − 04	75.31
c	3.11E − 02	3.28E + 02	2.58E − 04	75.31

Table A6
Lines 2 and 4 (with mutual coupling): system 2

Phase	R' (Ω/km)	Z_c (Ω)	τ (s)	Length (km)
b	7.07E − 01	7.35E + 02	10.04E − 04	183.14
c	5.13E − 02	4.69E + 02	6.99E − 04	183.14
a	3.13E − 02	3.32E + 02	6.29E − 04	183.14
a	3.06E − 02	3.18E + 02	6.22E − 04	183.14
b	3.03E − 02	2.78E + 02	6.18E − 04	183.14
c	3.01E − 02	2.75E + 02	6.17E − 04	183.14

Table A7
Source impedances for system 2

Bus		Actual value (Ω)
1	Z_0	0.333 + j14.366
	Z_1	1.083 + j35.255
3	Z_0	1.000 + j13.500
	Z_1	0.730 + j14.150

References

[1] *Proc Symp. Expert System Applications to Power Systems (ESAPS): 1st, Stockholm/Helsinki, Sweden/Finland, 1988; 2nd, Seattle, WA, USA, 1989; 3rd, Tokyo/Kobe, Japan, 1991; 4th, Melbourne, Australia, 1993.*

[2] M. Kezunović and P. Spasojević, An expert system for DFR file classification and analysis, *Proc. 4th Symp. Expert System Applications to Power Systems (ESAPS), Melbourne, Australia, 1993,* pp. 618–623.

[3] M. Kezunović, P. Spasojević, C.W. Fromen and D.R. Sevcik, An expert system for substation event analysis, *IEEE Trans. Power Delivery, 8* (4) (1993) 1942–1949.

[4] M. Kezunović, I. Rikalo, C.W. Fromen and D.R. Sevcik, Expert system reasoning streamlines disturbance analysis, *IEEE Comput. Appl. Power, 7* (2) (1994) 15–19.

[5] A. Cichocki and T. Lobos, Artificial neural networks for real time estimation of basic waveforms of voltages and currents, *Proc. IEEE Power Industry Computer Applications (PICA) Conf., Phoenix, AZ, USA, 1993,* pp. 357–363.

[6] A. Gosh and D.L. Lubkeman, The classification of power system disturbance waveforms using a neural network approach, *Proc. IEEE Transmission and Distribution Conf., Chicago, IL, USA, 1994,* pp. 323–329.

[7] T.S. Sidhu, H. Singh and M.S. Sachdev, Design, implementation and testing of an artificial neural network based fault direction discriminator for protecting transmission lines, *IEEE Trans. Power Delivery, 10* (2) (1995) 697–706.

[8] T. Dalstein and B. Kulicke, Neural network approach to fault classification for high speed protective relaying, *IEEE Trans. Power Delivery, 10* (2) (1995) 1002–1012.

[9] M. Kezunović, I. Rikalo and D.J. Šobajić, Neural network applications to real-time and off-line fault analysis, *Proc. Int. Conf. Intelligent Systems Applications to Power Systems, Montpellier, France, 1994,* EC2, 269 rue de la Garenne, 92024 Nanterre Cedex, France, pp. 29–36.

[10] M. Kezunović, I. Rikalo, C.W. Fromen and D.R. Sevcik, Automated fault analysis using neural network, *9th Annu. Conf. Fault and Disturbance Analysis, Texas A&M University, College Station, TX, USA, 1994.*

[11] M. Kezunović, Implementation framework of an expert system for fault analysis, *Proc. 3rd Symp. Expert System Applications to Power Systems (ESAPS), Tokyo/Kobe, Japan, 1991,* pp. 80–86.

[12] R.O. Duda and P.E. Hart, *Pattern Recognition and Scene Analysis,* Wiley, New York, 1973.

[13] *Electromagnetic Transients Program — Workbook,* Electric Power Research Institute, Palo Alto, CA, Sept. 1986.