Adaptive Multiple Fault Detection and Alarm Processing for Loop System With Probabilistic Network

Whei-Min Lin, *Member, IEEE*, Chia-Hung Lin, and Zheng-Chi Sun

***Abstract*—This paper presents the fault detection and alarm processing in loop system with fault detection system (FDS). FDS consists of adaptive architecture with probabilistic neural network (PNN). Training PNN uses the primary/backup information of protective devices to create the training sets. However, when network topology changes, adaptation capability becomes important in neural network applications. PNN can be retained and estimated effectively. With a looped system, computer simulations were conducted to show the effectiveness of the proposed system, and PNN's adapt network topology changes.**

***Index Terms*—Adaptation capability, fault detection system, probabilistic neural network.**

I. Introduction

POWER system protection is important for service reliability and quality assurance. Various faults may occur due to natural and artificial calamity. To reduce the outage duration and promptly restore power services, fault section estimate has to be done effectively and accurately with fault alarms. Dispatchers study the changed statuses of protection devices from the supervisory control and data acquisition (SCADA) system to identify the fault. Single and multiple faults could coexist with the failed operation of relays and circuit breakers, or with the erroneous data communication. It needs a long time to process a large number of alarms under various conditions involving multiple faults and many uncertainties. To cope with the problem, an effective tool is helpful for the fault section estimation and alarm processing.

Applications of artificial neural network have been presented in dealing with the fault diagnosis and alarm processing [1]–[7]. Many researchers have applied the ANN to fault section estimation [1]–[6], and alarm processing [7]. ANN is very useful owing to its parallel distributed process, training capability, and implicit knowledge representation. Many papers have presented the use of ANN for power system applications such as the use of multilayer back-propagation network (BPN) [2], [3]. Training BPN is time consuming and very slow without guaranteed global minimum. The global optimization methods were also proposed to solve the fault section estimation such as the Boltzmann machine [1]; however, the training process is still very time consuming. Another problem of multilayer network is that it is difficult to decide the number of layers and the number of hidden units in each layer. When network topology changes, the network has to retrain, and the time consuming process becomes a bottleneck in environment adaptation.

A robust estimation method must effectively deal with uncertainties in fault diagnosis. In reality, under severe situations such as typhoon, major storms, or a strong earthquake as the sometimes very hostile environment in Taiwan, power systems are very often under multiple attacks with probable device failures and communication errors. Furthermore, several switches might have to operate to change the network structure. Needless to say, the adapting capability and the performance in adapting itself to changes are crucial in dealing with the severe situation, where a major algorithm overhaul or repetitive training process will not be acceptable. Multiple fault detection with effective adaptation is strongly desirable to withstand certain noise levels and network topology changes [5]. Probabilistic neural network (PNN) [8], [9] was thus studied and proposed in this paper for fault section estimation and alarm processing. PNN can function as a classifier, to learn to place test exemplars into one of two or more categories. PNN has a number of input nodes equal to the number of predictor variables, and also has a number of hidden nodes equal to the number of training exemplars, with one hidden node assigned to each training exemplar. Output nodes of PNN is equal to the number of dependent variables whose values are being predicted. Another advantage of PNN is the single-pass network training stage without any iteration for adjusting weights.

When the environment changes, a completely different ANN architecture is required. PNN is easy to retrain with new data and adapt itself to architectural changes, such as network topology changes. A loop power system will be studied for example. This paper presents the algorithm of using PNN for a proposed fault detection system (FDS). Computer simulations will also be shown with test results provided.

II. Probabilistic Neural Network (PNN)

Architecture of the PNN is shown in Fig. 1. PNN contains three layers: the input layer, hidden layer, and one output layer. In the hidden layer, an activation function is applied to the distance measure between the unknown input and the training exemplar. For example [10], Fig. 1 is designed to classify an input vector into one of two categories with category 1 for "*fault*," and category 2 for "*No fault*." Input vector $\mathrm{X} = [x_1, x_2, x_3]$ is applied to input nodes I_1 through I_3. In the hidden layer, the network contains four nodes H_1 through H_4, corresponding to four

Manuscript received August 19, 2002.

The authors are with the Department of Electrical Engineering, National Sun Yat-Sen University, Kaohsiung 80424, Taiwan, R.O.C. (e-mail: wmlin@ee.nsysu.edu.tw).

Digital Object Identifier 10.1109/TPWRD.2003.820203

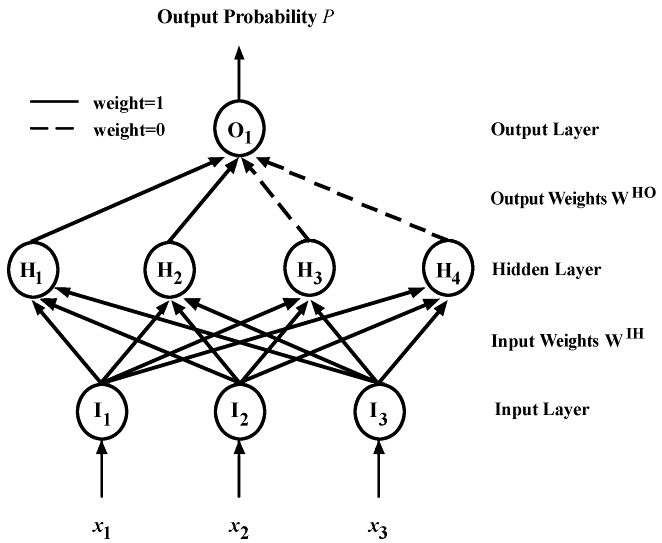


Fig. 1. Architecture of the PNN.

exemplars with weights connecting to the input nodes. Output weights are assigned 1 for "*category* 1" and 0 for "*category* 2." Weights between the hidden nodes and output node O_1 are designed to allow O_1 to compute the sum over the probabilities that correspond to only category 1. That is, O_1 sums the probabilities of H_1 and H_2 without probability $P = (\mathrm{H}_1 + \mathrm{H}_2)/(\mathrm{H}_1 + \mathrm{H}_2 + \mathrm{H}_3 + \mathrm{H}_4)$.

Expanding the architecture of PNN, we could have n input nodes I_1 through I_n, K hidden nodes (K training exemplars), and m output nodes O_1 through O_m. PNN Algorithm has two stages: the learning stage and recalling stages as delineated below.

A. Learning Stage

Step 1) For each training exemplar $\mathrm{X}(k)$, $k = 1, 2, \ldots, K$, create input weights W^{IH} between input node I_i and hidden node H_k by

$$w_{ki}{}^{\mathrm{IH}} = x_i(k) \quad (i = 1, 2, \ldots, n) \tag{1}$$

with $\mathrm{W}^{\mathrm{IH}} = [w_{ki}{}^{\mathrm{IH}}]_{p\times n}$ and $\mathrm{X}(k) = [x_1(k), \ldots, x_i(k), \ldots, x_n(k)]^{\mathrm{t}}$, $x_i(k) \in \{0, 1\}$.

Step 2) Output weights are 1 for category 1 and 0 for category 2. Create output weights W^{HO} between hidden unit H_k and output hidden O_j by

$$w_{kj}{}^{\mathrm{HO}} = \begin{cases} 1, & k \in \text{Category } 1 \\ 0, & k \in \text{Category } 2 \end{cases} \tag{2}$$

with $\mathrm{W}^{\mathrm{HO}} = [w_{kj}{}^{\mathrm{HO}}]_{p\times m} \cdot (j = 1, 2, \ldots, m)$.

where

K number of training exemplars;

n dimension of X;

m dimension of Y.

B. Recalling Stage

Step 1) get network weights W^{IH} and W^{HO};

Step 2) apply test vector $\mathrm{X} = [x_1, \ldots, x_i, \ldots, x_n]$ to the network;

Step 3) compute the probability of test vector X by Gauss activation function to the distance measure between the unknown test vector and all of the training exemplars

$$net_k = \sum_{i=1}^{n} \left(x_i - w_{ki}{}^{IH}\right)^2 \tag{3}$$

$$\mathrm{H}_k = \exp\left(-\frac{net_k}{2\sigma^2}\right) \tag{4}$$

where smoothing parameter $\sigma_1 = \sigma_2 = \ldots = \sigma_K = \sigma$ will be tested and discussed in a later section.

Step 4) compute the sum of the probabilities O_j by

$$O_j = \sum_{k=1}^{K} w_{kj}{}^{\mathrm{HO}} \mathrm{H}_k \quad (k \in \text{Category } 1) \tag{5}$$

where H_k must belong to *category* 1.

Step 5) normalize the sum of probabilities by dividing the sum over all H_k. Output probability P_j is

$$Prob\ P_j = \frac{\mathrm{O_j}}{\sum_{k=1}^{K} \mathrm{H}_k}. \tag{6}$$

III. Design of the Detection System

A. Protection Blocks

Protection blocks are used in this paper to define the protection zone for protective devices. Fig. 2 illustrates the concept of the protection blocks. There are 14 blocks consisting of two transformers (T1∼T2), 5 transmission lines (L1∼L5), and seven buses (bus 1∼bus 7). Each block contains its protective devices and each circuit breaker is included in two neighboring blocks. Protection blocks with protective devices are required to function immediately under abnormal conditions to avoid serious damage. These devices must also provide backup protection once the primary protection fails. In this paper, four types of protective schemes are considered: the line protection, bus protection, transformation protection, and backup protection.

B. Training Exemplar Creation

Protective devices should operate in a coordinated manner to reduce the risk of equipment damage. If primary protective devices fail to operate properly, backup protection must operate to clear the fault. It is important to properly coordinate the primary and backup protection system. For example, assuming a fault occurred on L2 (From bus 3 to bus 4), primary relay reacts to trip associated circuit breakers CB6 and CB7. If primary protection fails to operate, backup relays have to trip breakers CB4 and CB9. Various components have their own primary and backup protection scheme. Associate fault components with the statues of primary/backup protective devices would form particular symptomatic patterns. The possible fault events considered in this paper are:

- single and multiple faults;
- single fault with failed operation of relays or breakers;
- multiple faults with failed operation of relays or breakers;

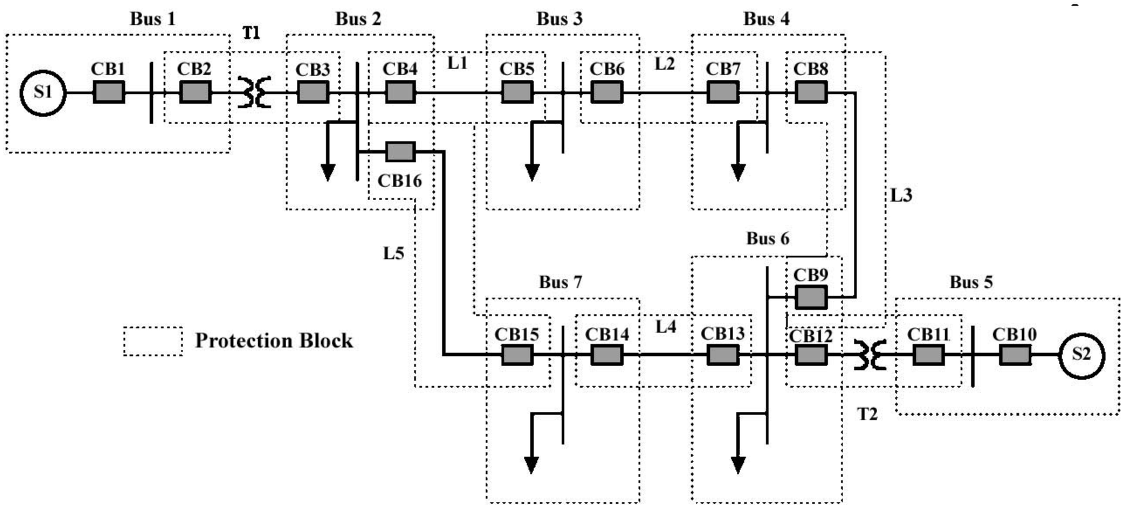


Fig. 2. Example of protection blocks.

- data communication with errors;
- faults with architectural changes.

Considering single and double faults, the total number of possible fault events for m components can be estimated by

$$\mathrm{Event}_m = \frac{m!}{1!(m-1)!} + \frac{m!}{2!(m-2)!}. \tag{7}$$

For each event, the states of protective devices and monitored components could be encoded as binary values of 1 or 0, with signal 1 for "*operation*" or "*fault*," 0 for "*not operational*," "*uncertain*," or "*No fault*." According to the symptomatic patterns, we can create training exemplars for PNN. In matrix W^{IH}, each row constitutes one exemplar, and each column represents one protective device. In matrix W^{HO}, each row represents a hidden node, and each column represents a monitored component. Architecture of the PNN will have

- the number of input nodes being equal to the number of protective devices;
- the number of hidden nodes being equal to the number of training exemplars;
- the number of output nodes being equal to the number of monitored components.

In the previous example, Fig. 2 is a looped system with two power sources. The closed dashed lines indicate protection blocks. Each block contains one component in addition to associated relays and two circuit breakers. The symbols of relays are defined by

- LR: primary protective relay of transmission lines;
- BR: primary protective relay of buses;
- TR: primary protective relay of transformers;
- RR: backup protective relay related to LR.

PNN acquires information of primary protective devices from SCADA. According to the information of protective devices, the training data for Fig. 2 are shown in Table I. Table I shows the number of protective devices and training exemplars for lines, buses, and transformers. For example, principal protective components of lines are illustrated in Fig. 3. The fault region will be extended by failed operation of protective devices. If a fault occurred on L1, and CB4 failed to trip, the backup relay RR2 and RR15 would operate to trip CB2 and CB15. In extreme conditions, power sources could be disconnected from the rest of the power system (S1/S2).

TABLE I
TRAINING DATA FOR THE DETECTION SYSTEM

5 line-segment protection: L1, L2, L3, L4, & L5		
Protective Type	Breakers and Relays	Fault type
Primary Protection	CB4,CB5,CB6,CB7,CB8,CB9,CB13, CB14,CB15,CB16,LR4,LR5,LR6,LR7, LR8,LR9,LR13,LR14,LR15,LR16	Normal 5 single faults 10 double faults
5 bus protection: Bus1, Bus2, Bus3, Bus4, Bus5, Bus 6, & Bus7		
Protective Type	Breakers and Relays	Fault type
Primary Protection	CB1,CB2,CB3,CB4,CB5,CB6,CB7,CB8, CB9,CB10,CB11,CB12,CB13,CB14,CB15, CB16,BR1,BR2,BR3,BR 4,BR5,BR6,BR7	Normal 7 single faults 21 double faults
2 transformer protection: T1 & T2		
Protective Type	Breakers and Relays	Fault type
Primary	CB2,CB3,CB11,CB12,TR1,TR2	Normal 2 single faults 1 double fault
Back-up information for Lines, Buses and transformers		
Protective Type	Breakers and Relays	Fault type
Back-up 1 Protection	RR2,RR3,RR4,RR5,RR6,RR7,RR8, RR9,RR11,RR12,RR13,RR14,RR15, RR16,LR5,LR7,LR9,LR13,LR15,S1,S2	14 failed relays or breaker
Back-up 2 Protection	RR2,RR3,RR4,RR5,RR6,RR7,RR8, RR9,RR11,RR12,RR13,RR14,RR15, RR16,LR4,LR6,LR8,LR14,LR16,S1,S2	14 failed relays or breaker

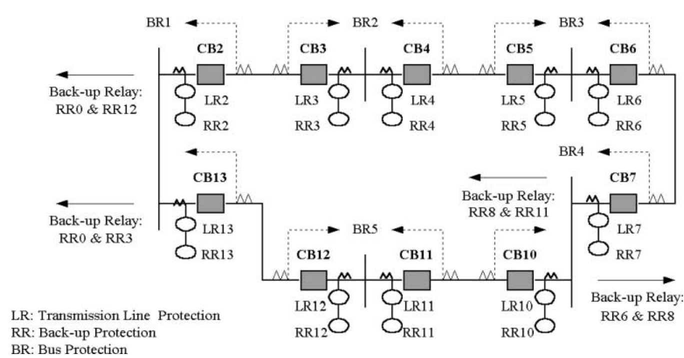


Fig. 3. Protective components of lines.

Information of fault alarms has some problems such as data loss, protection devices failures, and communication errors. Incoming alarms of normal and abnormal operation for various

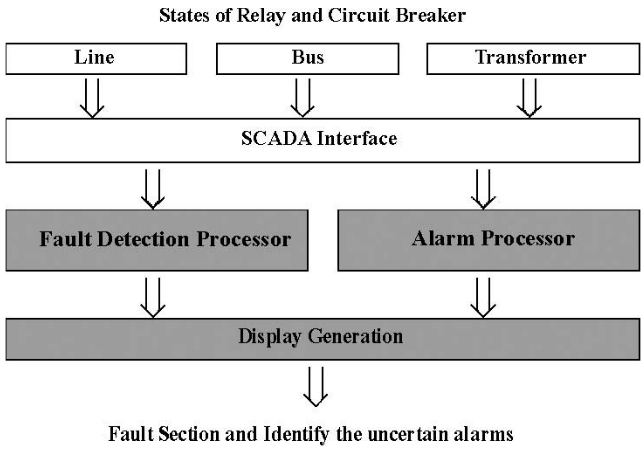


Fig. 4. Architecture of the FDS.

conditions could include possible fault events and events with protection devices failures. We can create training exemplars (input/output pairs) for alarm processing based on PNN. The K training exemplars are shown in Fig. 6

Alarm processing outputs the post-processing alarms, and compares the incoming alarms. The results will be provided to identify the failed protection devices and communication errors.

C. Architecture of the Fault Detection System (FDS)

Architecture of the proposed FDS based on PNN is shown in Fig. 4. FDS includes SCADA interface, fault detection processors (FDPs), alarm processor (AP), and display generation. In Fig. 4 [6], FDP has four fault processors including the line (DS-line), bus (DS-bus), transformer (DS-Tr), and backup (DS-backup 1/2) processors. Each processor processes the information of primary and backup protective devices. The output values of P are evaluated with FDP, where the values are between 0 and 1. A calculated P value close to 1 means "*fault*," and 0 means "*no fault*." Display generation will use these values to determine the fault status. Considering the possible device failure and erroneous data communications, a threshold value is designed for error-tolerance and to separate "*fault*" from "*no fault*," either a high value greater than 0.5, or a low value of less than 0.5 can be used. The medium value of 0.5 is used in this paper.

A sorting algorithm for the output P is used. There are many sorting algorithms, such as bubble sort, selection sort, and insertion sort, etc. In this paper, selection sort is used to find the maximum value in output P [11]. The vector P with m elements is sorted after $m-1$ steps. With the values greater than the threshold value, DS could identify the faulty component in display generation. In display generation, the output P_m classifies the statuses of monitored components according to the following threshold value:

- if $P_m \geq 0.5$, component m is faulty.
- if $P_m < 0.5$, component m is not faulty.

AP has three processors including the line (AP-line), bus (AP-bus), and transformer (AP-Tr) processors. AP outputs the vector of post-processing alarm $\mathrm{Alarm}_{\mathrm{post}}$, and compares with the vector of incoming alarms $\mathrm{Alarm}_{\mathrm{in}}$ as follows:

$$\mathrm{Alarm} = [\mathrm{Alarm}_{\mathrm{post}} \oplus (\mathrm{Alarm}_{\mathrm{post}} \odot \mathrm{Alarm}_{\mathrm{in}})] \quad (8)$$

TABLE II
ARCHITECTURES OF PNN FOR EACH FDP AND AP

FDP and AP	Network Size (Nodes)			Smoothing Parameter σ
	Input	Hidden	Output	
DS-Line	20	16	5	0.8
DS-Bus	23	29	7	0.5
DS-Tr	6	4	2	0.7
DS-Back-up 1/2	21	14	14	0.6
AP-Line	20	25	20	0.6
AP-Bus	23	35	23	0.5
AP-Tr	12	5	12	0.6

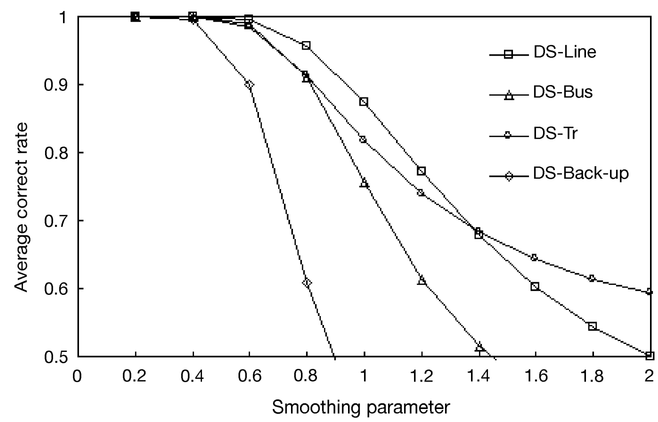


Fig. 5. Average output levels versus the smoothing parameters.

TABLE III
COMPARISON BETWEEN BAMN AND BPN

Neural Network	Network Size (Units)			Execution Time	Average Iteration
	Input	Hidden	Output		
BPN	20	40	5	>78min	>5,000
PNN	20	16	5	270ms	None

where

$$\mathrm{Alarm}_{\mathrm{post}} \in \{0,1\} \text{ and } \mathrm{Alarm}_{\mathrm{in}} \in \{0,1\}.$$
$$\oplus : \text{the Exclusive} - \text{OR operation.}$$
$$\odot : \text{the AND operation.}$$

In display generation, the vector of alarm is used to identify the abnormal signals, with 1 for "*abnormal*." If non backup relay operates to trip the circuit breaker, the result is regarded as communication errors. The four possible detected results are

- normal operation (of both relay and breaker);
- nonoperation (of relay);
- nontripping (of the breaker);
- communication error (on the relay or breaker).

With cause-effect training exemplars between fault components and operation of protection devices, the outputs of FDPs and APs are causal and coherent. Integrating the results, fault sections and abnormal alarms are identified. The results are provided to dispatchers for analysis to schedule proper maintenance and restoration for protection devices.

IV. SIMULATION RESULTS

A previous looped system is used for test. FDS has 14 protection blocks including five lines (L1∼L5), seven buses (bus 1∼bus 7), and two transformers (T1∼T2). Each block contains

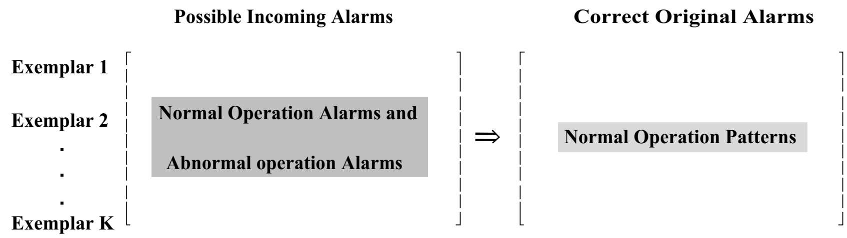


Fig. 6. K training exemplars

its protective relays and breakers. According to the proposed method, elements of every block will be included in one of the three processors–DS-line, DS-bus, and DS-Tr. Statuses of protective devices are provided by SCADA. DS-backup (backup 1/2) will be functional for auxiliary detection with statuses of possible backup protective devices. AP has three processors AP-line, AP-bus, and AP-Tr. Architectures of PNN for each FDP and AP are shown in Table II. All processors are designed on a Pentium III PC with 128-MB random-access memory (RAM) and Matlab software. To show the effectiveness of the proposed detection system, some testing cases were chosen for demonstration.

A. Case 1: Multiple Fault Detection

Case 1 shows triple faults on L1, L5, and bus 2. Primary relay LR4, LR5, LR15, and LR16 operated to trip CB4, CB5, CB15, and CB16. Relay BR2 operated to trip CB3, CB4, and CB16. DS-line and DS-bus detected the information of primary devices and indicated faults on L1, L5, and bus 2. AP outputs the post-processing alarms, and compares with incoming alarms. Results of AP-line and AP-bus show a normal operation of relay and breaker. The results are shown below

===== Fault Section

DS – Line : L1 $P = 0.9529$
L5 $P = 0.9529$
DS – Bus : Bus2 $P = 0.9813$

===== Alarm Identification
AP – Line : Normal operation
AP – Bus : Normal operation

B. Case 2: Single Fault With One Failed Circuit Breaker

Case 2 shows single fault on bus 3 with one CB failure. Primary relay BR3 operated to trip CB5, but failed to trip CB6. Backup relay RR7 will operate to trip CB7. DS-bus detected the information of protection devices and indicated single fault on bus 3. The results are

===== Fault Section

DS – Bus : Bus3 $P = 0.8807$

===== Alarm Identification
AP – Bus : CB6 Non – tripping

C. Case 3: Double Faults With One Failed Relay

Case 3 shows double faults on L1 and L5 with one failed relay. Relay LR4, LR15, and LR16 operated to trip breakers CB4, CB15, and CB16. Relay LR5 failed to operate, and backup relay RR7 will operate to trip CB7. DS-line and DS-backup detected the information and indicated double faults on L1 and L5. The results are

===== Fault Section

DS – Line : L5 $P = 0.9524$
DS – Back – up : L1 $P = 0.8800$

===== Alarm Identification
AP – Line : LR5 Non – operation
CB5 Non – tripping

D. Case 4: Data Communication With Error

A single fault on bus 3 with one communication error on breaker CB6 was tested. DS-bus detected the information of protection devices and indicated single fault on bus 3. The results are

===== Fault Section

DS – Bus : Bus3 $P = 0.8808$

===== Alarm Identification
AP – Bus : CB6 Communication error

E. Case 5: Test for Topological Changes

Training exemplars of FDS and AP could change owing to structural changes of the power system. Let there be a fault on L5, primary protection devices operated to clear the fault and remove L5. When the environment changes, old training exemplars must be modified by deleting the L5-related data, and retraining PNN with new training exemplars. The architectures of PNN will become

- DS-line: 16 input, four output, 11 hidden nodes, $\sigma_{\text{Line}} = 0.8$;
- DS-bus: 21 input, seven output, 29 hidden nodes, $\sigma_{\text{Bus}} = 0.5$;
- AP-line: 16 input, 16 output, 18 hidden nodes, $\sigma_{\text{Line}} = 0.6$;
- AP-bus: 21 input, 21 output, 35 hidden nodes, $\sigma_{\text{Bus}} = 0.5$.

PNNs are easy to adapt to new environment for structural changes. The networks will adjust easily to the new training exemplars without any iterative operation. Let double faults occurred on L1 and bus 2. New DS-line and DS-bus could resume the duty of detecting the information, and indicate double faults on L1 and bus 2. The results are

$$\begin{array}{|l|}\hline \text{===== Fault Section} \\ \qquad \text{DS} - \text{Line : L1} \qquad P = 0.9577 \\ \qquad \text{DS} - \text{Bus : Bus2} \qquad P = 0.8805 \\ \text{===== Alarm Identification} \\ \qquad \text{AP} - \text{Line : Normal operation} \\ \qquad \text{AP} - \text{Bus : Normal operation} \\ \hline \end{array}$$

F. Smoothing Parameter Test

Changing the smoothing parameter σ from 0.2 to 2.0, Fig. 5 shows the average output levels of each FDP versus the smoothing parameters. The average output level decreases for wider σ. Network performance is affected by the width of the Gauss function (PDF). As the width of Gauss function decreases, decision boundaries can become increasingly nonlinear. For very narrow Gauss function, the network approaches a nearest neighboring classifier.

G. Performance Reference

The proposed network is superior to other multilayer networks in many folds, such as the very fast learning and recalling speed, no iteration for updating weights, and no estimation for the number of layers and hidden nodes. With the predetermined training exemplars, the number of hidden nodes could be effectively determined. Table III shows the DS-line comparison chart of PNN and BPN for reference.

V. CONCLUSION

A fault detection system (FDS) with PNN has been developed in this paper. With the information provided by SCADA, FDPs were used to detection fault sections. Selection sort was used to sort the output to find values greater than the threshold for Display. APs were used to identify the abnormal alarms and were integrated for display in the display generation. Some advantages of the PNN are

- very fast learning and recalling process;
- no iteration for weight regulations in learning process;
- no predecision for the number of hidden layers and the number of hidden nodes in each layer. With the predetermined training exemplars, the number of hidden nodes could be effectively determined.
- limit number of training exemplars for training;
- adaptability for architectural changes.

Computer simulation shows that PNN-based fault detection system could be very effective in processing the fault information to aid dispatchers to detect the fault.

REFERENCES

[1] T. Oyama, "Fault section estimation in power system using Boltzmann machine," in *Proc. 2nd Forum Artificial Neural Network Applicat. Power Syst.* , Japan, 1993, pp. 1–7.

[2] H.-T. Yang, W.-Y. Chang, and C.-L. Huang, "A new neural network approach to on-line fault section estimation using information of protective relays and circuit breakers," *IEEE Trans. Power Delivery*, vol. 9, pp. 220–229, Jan. 1994.

[3] T. Dalstein, T. Friedrich, B. Kulicke, and D. Sobajic, "Multi neural network based to fault area estimation for high speed protective relays," *IEEE Trans. Power Delivery*, vol. 11, pp. 740–747, Apr. 1996.

[4] J. C. S. de Souza and M. A. P. Rodrigues, "Fault location in electrical power systems using intelligent systems techniques," *IEEE Trans. Power Delivery*, vol. 16, pp. 59–67, Jan. 2001.

[5] T. Bi, Y. Ni, and F. F. Wu, "Distributed adaptive fault section estimation system for large-scale power networks," in *Proc. Power Eng. Soc. Winter Meeting*, vol. 2, 2002, pp. 1350–1353.

[6] W.-M. Lin, C.-H. Lin, Z.-C. Sun, and M.-T. Tasy, "Neural network approach to fault section detection in loop system," in *Proc. 22th Symp. Elect. Power Eng.*, Taiwan, R.O.C., Nov. 2001, pp. 1728–1732.

[7] R.-L. Jiang and Q.-X. Lin, "A neural network approach to alarm processing of power system in mass rapid transit," in *Proc. 20th Symp. Elect. Power Eng.*, Taiwan, R.O.C., Nov. 1999, pp. 1247–1251.

[8] D. F. Specht, "Probabilistic neural network for classification, mapping, or associative memory," in *Proc. IEEE Int. Conf. Neural Network*, vol. 1, San Diego, CA, July 1988, pp. 525–532.

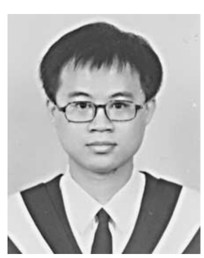
[9] ——, "A general regression neural network," *IEEE Trans. Neural Networks*, pp. 568–576, 1991.

[10] E. J. Rzempoluck, *Neural Networks Data Analysis Using Simulnet*. New York: Springer-Verlag, 1998.

[11] S. Lipschutz, *Theory and Problems of Data Structures*. New York: McGraw-Hill, 1986.

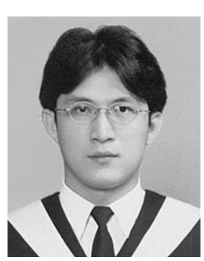
Whei-Min Lin (M'87) was born on October 3, 1954. He received the B.S.E.E. degree from the National Chao-Tung University, the M.S.E.E. degree from the University of Connecticut, Storrs, and the Ph.D. degree in electrical engineering from the University of Texas, in 1985.

Currently, he is with National Sun Yat-Sen University, Kaohsiung, Taiwan, R.O.C., where he has been since 1991. He was a visiting researcher with Chung-Hwa Institute for Economic Research, Taiwan. He was with Control Data Corp. in 1986 and worked with Control Data Asia in 1989. His research interests are GIS, distribution system, SCADA, automatic control system, and artificial intelligence applications in power systems.



Chia-Hung Lin was born on August 30, 1974. He received the B.S.E.E. degree from the Tatung Institute of Technology in 1998 and the M.S.E.E. degree from the National Sun Yat-Sen University, Kaohsiung, Taiwan, R.O.C. He is currently pursuing the Ph.D. degree at the National Sun Yat-Sen University.

His research interests are fault diagnosis in power system and neural network computing.



Zheng-Chi Sun was born on August 25, 1977. He received the B.S.E.E. degree from the Kun-Shon University of Technology in 2000, and the M.S.E.E from the National Sun Yat-Sen University, Kaohsiung, Taiwan, R.O.C., in 2002.

His research interests are fault diagnosis in distribution system and neural network computing.