A Novel Fuzzy Neural Network Based Distance Relaying Scheme

P. K. Dash, A. K. Pradhan, and G. Panda

***Abstract*—This paper presents a new approach to distance relaying using fuzzy neural network (FNN). The FNN can be viewed either as a fuzzy system, a neural network or fuzzy neural network. The structure is seen as a neural network for training and a fuzzy viewpoint is utilized to gain insight into the system and to simplify the model. The number of rules is determined by the data itself and therefore smaller number of rules is produced. The network is trained with back propagation algorithm. A pruning strategy is applied to eliminate the redundant rules and fuzzification neurons, consequently a compact structure is achieved. The classification and location tasks are accomplished by using different FNN's. Once the fault type is identified by the FNN classifier the selected fault locating FNN estimates the location of the fault accurately. Normalized peaks of fundamental voltage and current waveforms are considered as inputs to all the networks and an additional input derived from dc component is fed to fault locating networks. The peaks and dc component are extracted from sampled signals by the EKF. Test results show that the new approach provides robust and accurate classification/location of faults for a variety of power system operating conditions even with resistance in the fault path.**

I. INTRODUCTION

DISTANCE relaying techniques based on the measurement of impedance at the fundamental frequency between the fault location and the relaying point have attracted widespread attention. The sampled voltage and current data at the relaying point are used to locate and classify the type of fault involving the line with or without fault resistance present in the fault path. The accuracy of the fault classification and location also depends on the amplitude of the DC offset and harmonics in comparison to the fundamental component. Fourier transforms, Differential equations, Waveform modeling, and Kalman filters are some of the techniques used for fault detection and location calculations.

In most of the power system protection techniques the system states are defined by identifying the pattern of associated voltage and current waveforms. Therefore, the development of adaptive protection can be essentially treated as a pattern recognition problem [1]. A neural network which can perform pattern matching task has a large number of highly interconnected processing elements (nodes) that demonstrate the ability to learn and generalize from training patterns. Distributed representation and strong learning capabilities are the major features of neural network. Fuzzy logic systems on the other hand base their decisions on inputs in the form of linguistic variables derived from membership functions which are formulas used to determine the fuzzy set to which a value belongs and the degree of membership in that set. The variables are then matched with the specific linguistic IF–THEN rules and the response of each rule is obtained through fuzzy implication. To perform compositional rule of inference, the response of each rule is weighted according to the impedance or degree of membership of its inputs and the centroid of the response is calculated to generate the appropriate output [2]–[4]. Some of the neural network and fuzzy logic applications in power system protection are included in [1], [5]–[11].

Neural network has the shortcoming of implicit knowledge representation, whereas, fuzzy logic systems are subjective and heuristic. The determination of fuzzy rules, input and output scaling factors and choice of membership functions depend on trial and error that makes the design of fuzzy logic system a time consuming task. These drawbacks of neural network and fuzzy logic systems are overcome by integrating the learning capabilities of neural network to the robustness of fuzzy logic systems in the sense that fuzzy logic concepts are embedded in the network structure. It also provides a natural framework for combining both numerical information in the form of input/output pairs and linguistic information in the form of IF–THEN rules in a uniform fashion. One such application to power system fault classification problem is found in [12]. However, this approach needs sequence components to solve the classification problem and the training is carried out using the information from both designer's experiences and sample data sets. The other drawback of this approach is that the number of fuzzy rules increases exponentially with respect to inputs and as a consequence 17 rules are framed for 3 inputs only [12]. Again, three different neuro-fuzzy networks in series are proposed there to classify the type of fault.

In this paper a simple neural network is used to implement a fuzzy-rule-based classifier of a power system from input/output data. The FNN model can be viewed either as a fuzzy system, a neural network or a fuzzy-neural system. The structure is seen in neural viewpoint for training and fuzzy viewpoint is utilized to gain insight into the system and to simplify the model. Unlike earlier approach [12], in this strategy the number of rules needed is determined by the data itself and consequently a smaller number of rules are produced. The network is trained using back propagation algorithm. To have a compact output structure, a pruning strategy eliminates the redundant rules and fuzzification neurons.

The classification and location tasks for distance relaying scheme in this work are accomplished by using different FNN's. All the networks use normalized peaks of fundamental component of current and voltage waveforms of the three phases as

Manuscript received August 10, 1999.

The authors are with the Department of Electrical Engineering, Regional Engineering College, Rourkela, India.

Publisher Item Identifier S 0885-8977(00)08128-0.

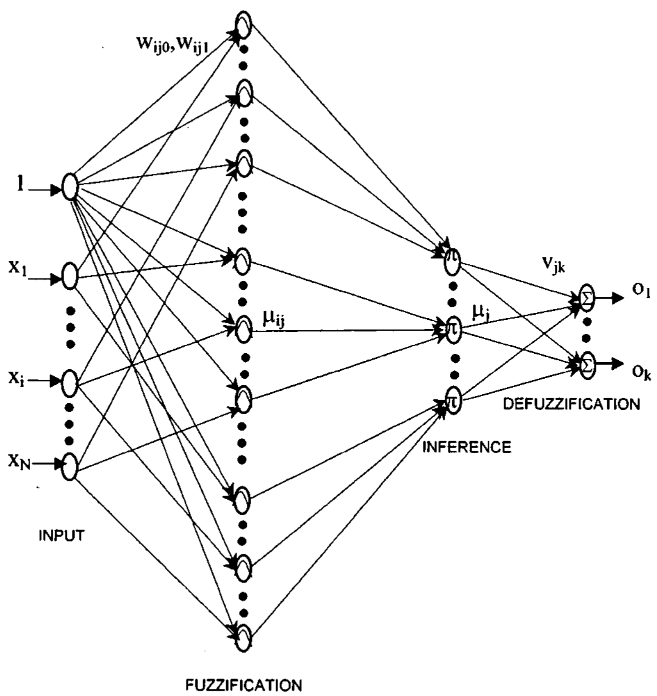

Fig. 1. The architecture of the fuzzy neural network.

inputs. Besides an additional input derived from decaying dc component is fed to the fault locating FNN's. The extraction of these components is carried out using an extended Kalman filter [13] suitably modeled to include decaying dc, third and fifth harmonics along with fundamental. The classification scheme is simpler and accurate fault classification is achieved in most of the fault types, the data of which are presented to the network. An EMTDC program [14] is used to generate fault data with different system conditions, source impedance, fault inception angle and fault resistance values. After the fault is successfully classified, the fault locator FNN is activated. The fault locator block comprises four FNN's, one each for the category of faults LG, LL, LLG and LLL. This approach provides accurate fault distance from the relaying point. Several test results are given in the paper to highlight the effectiveness of this new approach.

II. The Fuzzy Neural Network

Fig. 1 shows the architecture of the fuzzy neural network, comprising by input, fuzzification, inference and defuzzification layers. Further the network can be visualized as consisting of N inputs, with N neurons in the input layer and R rules, with R neurons in the inference layer. There are $N \times R$ neurons in the fuzzification layer and K neurons for output layer. The signal propagation and basic function in each layer of the FNN is introduced in the following.

The input layer consists of x_i, $i = 1, 2, \cdots, N$, along with unity. Each neuron in the fuzzification layer represents a fuzzy membership function for one of the input variables. The activation function used in this layer is $f(net_{ij}) = \exp(-|net_{ij}|^{l_{ij}})$ and the input to these neurons $net_{ij} = w_{ij1}x_i + w_{ij0}$, with w_{ij1} and w_{ij0} being the connecting weights between input layer and fuzzification layer.

Thus, the output of the fuzzification layer becomes

$$\mu_{ij}(x_i) = \exp(-|w_{ij1}x_i + w_{ij0}|^{l_{ij}}) \tag{1}$$

Where μ_{ij} is the value of fuzzy membership function of the ith input variable corresponding to the jth rule.

Each node j in the inference layer is denoted by π, which multiplies the input signals and the output of the node becomes the result of product. Therefore, the output of the layer becomes

$$\rho_j(x_1, x_2 \cdots x_N) = \prod_i^N \mu_{ij}(x_i) \tag{2}$$

With v_{jk} being the output action strength of the kth output associated with the jth rule and utilizing weighted sum defuzzification, the network output

$$\begin{aligned} o_k(x_1, x_2, \cdots x_N) &= \sum_j^R v_{jk}\rho_j(x_1, x_2, \cdots x_N) \\ &= \sum_j^R v_{jk} \prod_i^N \exp(-|w_{ij1}x_i + w_{ij0}|^{l_{ij}}) \end{aligned} \tag{3}$$

A. Training

Back propagation (BP) algorithm, which is the most popular method for neural network design, is being exploited to update parameters of the Fuzzy Neural network.

The error function E of the network be

$$E = \tfrac{1}{2}\sum_k (t_k - o_k)^2 \tag{4}$$

Where t_k is the desired output in the kth output node.

The parameter updation equations for the weights between the inference and output layers are:

$$v_{jk}(n) = v_{jk}(n-1) + \Delta v_{jk}(n) \tag{5}$$

Where

$$\Delta v_{jk}(n) = \eta\delta_k y_v + \alpha\Delta v_{jk}(n-1) \tag{6}$$

and n is the iteration count, $\delta_k = t_k - o_k$, $y_v = \rho_j$, η is the learning rate and α is momentum term.

For the weights between the input and fuzzification layer

$$w_{ij1}(n) = w_{ij1}(n-1) + \Delta w_{ij1}(n) \tag{7}$$

Where

$$\Delta w_{ij1}(n) = \eta\delta_{ij}y_1 + \alpha\Delta w_{ij1}(n-1) \tag{8}$$

$$\delta_{ij} = \rho_j \sum_k \delta_k v_{jk} \Big/ \mu_{ij}$$

and

$$y_1 = -l_{ij}x_i\mu_{ij}|w_{ij1}x_i + w_{ij0}|^{l_{ij}-1} \tag{9}$$

Similarly,

$$\begin{aligned} w_{ij0}(n) &= w_{ij0}(n-1) + \Delta w_{ij0}(n) \\ \text{where } \Delta w_{ij0}(n) &= \eta\delta_{ij}y_0 + \alpha\Delta w_{ij0}(n-1) \\ \mathrm{y}_0 &= -l_{ij}\mu_{ij}|w_{ij1}x_i + w_{ij0}|^{l_{ij}-1} \end{aligned} \tag{10}$$

The fuzzy membership function parameter is updated as

$$l_{ij}(n) = l_{ij}(n-1) + \Delta l_{ij}(n) \tag{11}$$

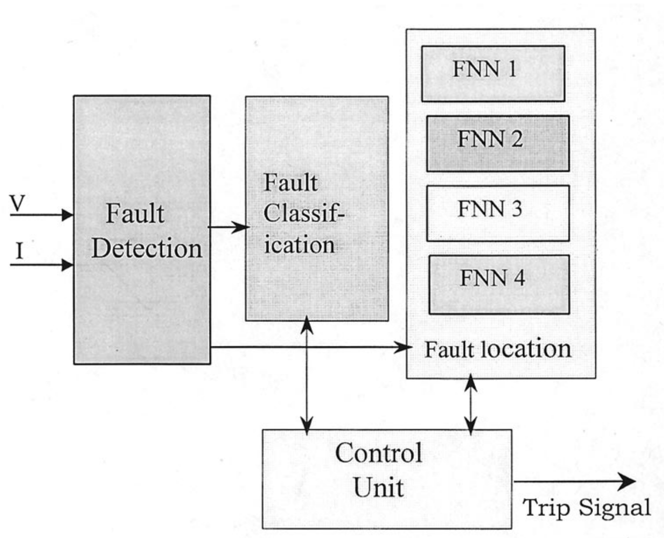

Fig. 2. The proposed distance protection scheme.

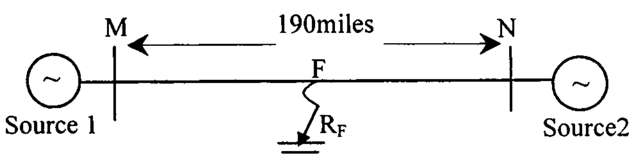

Fig. 3. 230 kV-transmission system.

Where

$$\Delta l_{ij}(n) = \eta \delta_{ij} y_i + \alpha \Delta l_{ij}(n-1)$$

and

$$y_i = -\mu_{ij} \log_e(|w_{ij1}x_i + w_{ij0}|)|w_{ij1}x_i + w_{ij0}|^{l_{ij}} \qquad (12)$$

During training, the number of rules is increased from 1 till a satisfactory performance of the network is found. The learning rate "η" which controls the rate of convergence initially set to 0.2 and is reduced gradually to 0.01 and the momentum constant "α," added to speed up the training and avoid local minima, is kept at 0.6 throughout. The initial weights are randomly selected in the interval $[-1, +1]$ and l_{ij} is initialized to 2. The number of iteration is set to 5000 in all cases. The training is continued till $E(n) < \varepsilon$ at all points for a window length of 100 or the number of iteration reaches its maximum. The value of ε is taken to be $1 * e^{-4}$ during training.

B. Pruning Strategy

Even if a single fuzzy membership function is near zero over its input range, the output of the corresponding rule becomes close to zero. As this rule does not contribute to the network performance, the rule should be pruned. Further, with multiple inference a fuzzy membership function having close to unity over its input range contributes negligibly to the network output. This neuron can be also eliminated without hampering the network performance. By removing these redundant rules and neurons form the structure a compact form is achieved. To implement this technique, we run the trained network with the same training sets once and see the outputs in inference and fuzzification layers. In the event of such situations as described above exist, corresponding neurons are pruned and then network performance is studied.

TABLE I
INCEPTION ANGLE $60°$ (FAULT AT 15% OF LINE)

Fault type	R_f=0Ω			R_f=10Ω		
	a	b	c	a	b	c
ag	1.0335	0.0635	0.0658	1.0353	0.0587	0.0633
bg	0.0690	1.1212	0.000	0.0702	1.1253	-0.0028
cg	0.0450	0.0415	1.0672	0.0405	0.0439	1.0693
ab	0.9388	1.0660	-0.0128	0.9043	1.0519	-0.0118
bc	0.0202	0.9881	0.9458	-0.0120	0.9941	0.9577
ca	0.9140	0.0490	0.9625	0.8904	0.0310	0.9557
abc	0.9740	0.9924	0.9334	0.9410	0.9828	0.9241

TABLE II
INCEPTION ANGLE $45°$ (FAULT AT 40% OF LINE)

Fault type	R_f=0Ω			R_f=10Ω		
	a	b	c	a	b	c
ag	1.0826	0.3758	0.0528	1.0738	0.3340	0.1518
bg	0.0521	1.1055	0.0970	0.0620	1.1039	0.0437
cg	-0.0153	0.3103	1.0824	-0.0143	0.2944	1.0708
ab	1.1079	1.0847	0.0818	1.0882	1.0838	0.0875
bc	-0.0432	1.0485	1.0371	-0.0416	1.0500	1.0407
ca	0.9975	0.3706	1.0047	1.0146	0.0540	1.0240
abc	1.0640	0.9978	0.9590	1.1231	0.9774	0.9609

TABLE III
INCEPTION ANGLE $30°$ (FAULT AT 65% OF LINE)

Fault type	R_f=0Ω			R_f=10Ω		
	a	b	c	a	b	c
ag	0.9932	0.0437	0.0386	0.9492	0.0465	0.0403
bg	0.0133	0.9845	0.0336	0.0132	0.9607	0.0348
cg	0.0582	0.0187	0.9734	0.0687	0.0192	0.9453
ab	1.0503	1.0435	-0.0360	1.0334	1.0023	-0.0344
bc	-0.0587	1.0421	1.0426	-0.0641	1.0254	1.0338
ca	1.0356	-0.0430	1.0383	1.0129	-0.0371	1.0170
abc	0.9943	0.9936	0.9838	0.9940	0.9936	0.9836

TABLE IV
INCEPTION ANGLE $90°$ (FAULT AT 80% OF LINE)

Fault type	R_f=0Ω			R_f=10Ω		
	a	b	c	a	b	c
ag	1.0371	0.1903	0.4024	1.0207	0.1883	0.3157
bg	0.3504	0.9428	0.0333	0.03136	0.9300	0.0458
cg	0.1502	0.1997	0.9358	0.1572	0.1836	0.9142
ab	1.0834	0.8597	0.1113	1.0861	0.7838	0.0893
bc	0.2549	1.0252	0.7449	0.0816	1.0507	0.5764
ca	1.0341	0.1654	1.0337	0.9812	0.1105	1.0362
abc	1.0234	0.9516	0.9431	1.0256	0.9730	0.9404
abg	1.0827	0.9240	0.1026	1.0817	0.8895	0.1035
bcg	0.2398	1.0381	0.7687	0.2310	1.0465	0.7273
cag	1.0264	0.1686	1.0317	1.0222	0.1637	1.0320

TABLE V
PHASE "a" TO GROUND FAULT AT DIFFERENT LOCATIONS WITH HIGH FAULT RESISTANCE ($R_f = 100\ \Omega$)

Dist(%) → / Output ↓	10	20	30	40	50	60	70	80
a	0.7358	0.7670	0.7692	0.7260	0.6704	0.5927	0.5101	0.5092
b	0.1477	0.1550	0.1551	0.1652	0.1910	0.2107	0.1826	0.1429
c	0.1554	0.1453	0.1449	0.1432	0.1394	0.1383	0.1624	0.2048

III. The Distance Protection Scheme

The proposed transmission system protection scheme utilizing Fuzzy Neural Network is shown in Fig. 2. On the occurrence of a fault, the fault detection unit activates the fault classification unit. The classification block consists of an FNN to select the phases involved with the fault accurately and a ground detection unit [15] running in parallel with the FNN completes the classification task of the distance relay. Once fault is classified, the control unit fires the proper fault location FNN. The fault location unit comprises of four FNN's, one network each for the four categories of fault (LG, LL, LLG and LLL). Thereafter, the control block derives the decision of trip or no-trip from the output signals of classifier and locator units.

For the application of the proposed distance protection strategy based on FNN's, a 230 kV, 190 miles transmission system as shown in Fig. 3 is considered. The EMTDC software package is used to simulate the sample power system and input, output pair of training and testing the different networks are generated. For the purpose, an extended Kalman filter [13] is used in this work to extract the fundamental and dc components from the sampled current and voltage signals. A sampling rate of 1 kHz in a 50 Hz power system is considered for the purpose.

IV. Training and Testing of FNN Classifier

Distance relaying algorithms use the fundamental components of voltage and current signals available at the relaying point to derive the trip decision during faulty conditions of the power network. The idea of phasors is being integrated with fuzzy neural network to derive a robust fault classifier for the protection of a transmission system. Normalized values of post fault peaks of fundamental components of voltages and currents of the three phases are considered as input vector for the network. These peaks are estimated from sampled values of the current/voltage signal with EKF within half a cycle after fault inception. The FNN network consists of three outputs representing "a," "b," "c" phases. During training these outputs are assigned "1" or "0" considering whether the fault is involved with that phase or not (for example, b-phase to ground fault case bg' the output will be assigned 0 1 0). The training sets are 49 in numbers which include data for 10%, 40%, 60% and 80% fault location for different fault inception angles and at different conditions of the system and seven types of shunt faults (ag, bg, cg, ab, bc, ca, abc). The number of rules is increased from 1 during the training process till satisfactory response of the network is derived. With BP algorithm the network is trained and finally pruning strategy is applied. The ultimate network

TABLE VI
$Index1$ VALUES FOR FAULT AT 10% OF THE LINE

Fault Type	*Index1*
ab	0.0036
abg(R_f=0Ω)	0.4771
abg(R_f=10Ω)	0.4592
abg(R_f=100Ω)	0.2104

structure becomes, 6 inputs, 4 rules, 23 fuzzification neurons and 3 outputs.

The performance of the above network is tested using voltage and current data of the power system during various types of shunt faults at different locations, inception angles and prefault conditions of the system. Table I–IV present some of the classification test results for the faulted transmission line. Table I shows the performance of FNN for different fault types at 15% of line for 60° inception angle and at a different voltage level of the sources for $R_f = 0\ \Omega$ and $R_f = 10\ \Omega$. The respective values in a, b, c columns for "bc" case with $R_f = 0\ \Omega$; "a" $= 0.0202$, "b" $= 0.9881$ and "c" $= 0.9458$ depict that the phases associated with the fault are "b" and "c" only. This classification approach takes a particular phase to be involved with fault if its corresponding value is greater than a threshold value of 0.5 else it categorizes the phase to be "undisturbed." For a similar condition as used in Table I, except the source impedance ratio changed to 90 from 1, Table II, provides the fault classification results for different faults at 40% of the line. Table III shows fault classification for a different condition of load angle ($=$20°) at $30°$ inception angle and at 60% of the line whereas Table IV presents for a $90°$ inception angle and a fault at 80% of the line. These results demonstrate the suitability of the network even for the untrained categories of fault; "abg," "bcg" and "cag" etc. which are included in Table IV (rows 8–10). To study the performance of the network a high fault path resistance (100 Ω) is considered. Table V shows the test results for single line to ground fault at different location with $R_f = 10\ \Omega$. The FNN also classifies correctly for other types of fault with such a high fault resistance. Observation of all test results ascertains that the FNN performs excellent even at different inception angle, fault resistance, fault location and prefault loading conditions.

A. Ground Detection

Usually it is not possible to identify ground only from peaks of fundamental components of voltages and currents (the input

TABLE VII
FAULT LOCATION DISTANCE ESTIMATED BY FNN1 (LG)

Dist.(%)→	15		25		35		45		55		65		75	
$R_f(\Omega)$ →	0	100	0	100	0	100	0	100	0	100	0	100	0	100
Error(%)→	5.12	2.94	6.51	3.23	2.92	5.47	4.45	3.07	3.77	5.46	1.98	0.54	7.71	7.63

TABLE VIII
FAULT LOCATION DISTANCE ESTIMATED BY FNN2 (LL)

Dist.(%)→	15		25		35		45		55		65		75	
$R_f(\Omega)$ →	0	100	0	100	0	100	0	100	0	100	0	100	0	100
Error(%)→	4.25	3.69	7.68	6.24	5.21	2.96	1.16	1.18	2.26	0.03	3.55	2.96	7.46	7.75

TABLE IX
FAULT LOCATION DISTANCE ESTIMATED BY FNN3 (LLG)

Dist.(%)→	15		25		35		45		55		65		75	
$R_f(\Omega)$ →	0	100	0	100	0	100	0	100	0	100	0	100	0	100
Error(%)→	3.55	2.61	7.66	5.11	2.68	2.74	1.44	0.71	0.34	1.25	0.87	3.59	5.58	7.91

TABLE X
FAULT LOCATION DISTANCE ESTIMATED BY FNN4 (LLL)

Dist.(%)→	15		25		35		45		55		65		75	
$R_f(\Omega)$ →	0	100	0	100	0	100	0	100	0	100	0	100	0	100
Error(%)→	3.37	3.22	7.55	7.50	6.96	6.63	3.59	2.29	2.74	0.67	1.58	0.39	3.36	3.32

vector to FNN classifier). Therefore, the ground detection task is not included in the FNN classifier. In reference [15] for detecting the involvement of ground during a fault, a zero sequence current based indicator of the type

$$Index1 = \frac{|Ia + Ib + Ic|}{\mathrm{median}(|Ia|, |Ib|, |Ic|)}$$

is proposed. Here Ia, Ib and Ic are the current phasors of the three phases at the relaying end. The phasors are estimated by the EKF and the corresponding $Index1$ value is calculated. When the $Index1$ value exceeds the threshold value of 0.05, it indicates the involvement of fault with ground. The ground detection is carried out in conjunction with the FNN calculations. Test results showing the values of $Index1$ for "a" phase to "b" faults at a distance of 10% of the line are presented in Table VI.

V. TRAINING AND TESTING OF FNN's FOR FAULT LOCATION

Once the fault is classified, the control unit activates the correct fault locating FNN. For fault location task, an FNN is proposed for each category of fault. The FNN fault locator calculates the normalized distance of the fault point from the relaying point. In all the four FNN locators the input vector consists of normalized peaks of fundamental components of current and voltage and the normalized value of $|I_{\mathrm{dc}}/\tau|$. Where I_{dc} is the dc component of a faulty phase current waveform at fault inception and τ is its decay rate as estimated by the EKF's within half a cycle of fault inception. In the case of LG fault locator, the first and second elements of input vector should be the corresponding values of faulty phase current and voltage, respectively, whereas for LL and LLG locators the first four input elements are the corresponding values of faulty phases. The training sets of the FNN's include fault data for different fault inception angles, prefault conditions, fault path resistances (0 Ω, 50 Ω and 100 Ω) and at different fault distances (10–80% at a step of 10%). The total number of such sets is 56 for all four FNN's. The networks are trained by BP algorithm and pruning strategy. The final structure of FNN1 (LG) locator becomes, 7 inputs, 6 rules, 39 fuzzification neurons and 1 output. Similarly for FNN2 (LL) the number of rules and fuzzification neurons are 7 and 47, respectively [for FNN3 (LLG) 7, 47 and FNN4 (LLL) 7, 48, respectively].

Table VI shows some of the test results for FNN1 locator (LG) at a different condition of the power system, at 60° of inception angle, for both without and with 100 Ω of fault resistance cases and at different locations of the fault. Similar results for LL, LLG and LLL FNN fault locators are shown in Table VII–IX. In all the test cases for the networks which include different fault inception angles, different locations of fault, various prefault conditions (including the source capacity) and different fault resistance values, the maximum error found is less than 8%. The percentage error is computed as

$$error(\%) = \frac{|actual\ \mathrm{distance} - calculated\ \mathrm{distance}|}{protected\ line\ length} * 100$$

The FNN's calculate the fault distance within 80% of the line with high accuracy and enhances the performance of distance relaying scheme.

VI. Conclusions

An efficient distance relaying scheme based on FNN is proposed in this paper. Both the classification and location objectives are carried out by different FNN's. The FNN classifier uses normalized peaks of voltage and current waveforms as input whereas the fault locating FNN's, in addition to the normalized peaks require a ratio derived from dc component of one of the current waveforms considered as input. The peaks as well as the dc component are estimated by the EKF from the sampled data. The networks are trained by BP algorithm and pruning strategy is applied to eliminate the nodes which do not result in any change of the network output. The trained networks are capable of providing fast and precise classification and location of fault for a variety of system conditions, different inception angles and fault resistances.

References

[1] M. Sanaye-Pasand and O. P. Malik, "High speed transmission system directional protection using an elman network," *IEEE Trans. on Power Delivery*, vol. 13, no. 4, pp. 1040–1045, 1998.

[2] C. T. Lin and C. S. G. Lee, "Neural-network-based fuzzy logic control and design system," *IEEE Trans. on Computers*, vol. 40, no. 12, pp. 1321–1336, 1991.

[3] S. Horikawa, T. Furuhashi, and Y. Uchikawa, "On fuzzy modeling using fuzzy neural networks with the back-propagation algorithm," *IEEE Trans. on Neural Networks*, vol. 3, no. 5, pp. 801–806, 1992.

[4] Y. Lin and G. A. Cunningham, "A new approach to fuzzy-neural system modeling," *IEEE Trans. on Fuzzy Systems*, vol. 3, no. 2, pp. 190–197, 1995.

[5] T. S. Sidhu, H. Singh, and M. S. Sachdev, "Design, implementation and testing of an artificial neural network based fault direction discrimination for protecting transmission lines," *IEEE Trans. on Power Delivery*, vol. 10, no. 2, pp. 697–706, 1995.

[6] T. Dalstein and B. Kulicke, "Neural network approach to fault classification for high speed protective relaying," *IEEE Trans. on Power Delivery*, vol. 10, no. 2, pp. 1002–1011, 1995.

[7] Y. H. Song, A. T. Johns, and Q. Y. Xuan, "Artificial neural network based protection scheme for controllable series-compensated EHV transmission lines," *IEE Proc. Gen. Trans. Disbn*, vol. 143, no. 6, pp. 535–540, 1996.

[8] Y. H. Song, Q. Y. Xuan, and A. T. Johns, "Protection scheme for EHV transmission systems with thyristor controlled series compensation using radial basis function neural networks," *Electric Machines and Power Systems*, pp. 553–565, 1997.

[9] ——, "Adaptive distance protection of double-circuit lines using artificial neural networks," *IEEE Trans. on Power Delivery*, vol. 12, no. 1, pp. 97–105, 1997.

[10] D. V. Coury and D. C. Jonge, "Artificial neural network approach to distance Ppotection," *IEEE Trans. on Power Delivery*, vol. 13, no. 1, pp. 102–108, 1998.

[11] A. Ferrero, S. Sangiovanni, and E. Zappitelli, "A fuzzy-set approach to fault—Type identification in digital relaying"," *IEEE Trans. on Power Delivery*, vol. 10, no. 1, pp. 169–175, 1995.

[12] H. Wang and W. W. L. Keertipala, "Fuzzy-neuro approach to fault classification for transmission line protection," *IEEE Trans. on Power Delivery*, vol. 13, no. 4, pp. 1093–1103, 1998.

[13] T. W. Hilands and S. C. A. Thomopoulos, "Nonlinear filtering methods of harmonic retrieval and model order selection in Gaussian and non-Gaussian noise," *IEEE Trans. on Signal Processing*, vol. 45, no. 4, pp. 983–994, 1997.

[14] Manitoba HVDC Research Center, "EMTDC—Electromagnetic transients simulation program,", 1988.

[15] M. Akke and J. S. Thorp, "Some improvements in the three-phase differential equation algorithm for fast transmission line protection," *IEEE Trans. on Power Delivery*, vol. 13, no. 1, pp. 66–72, 1998.

P. K. Dash is a Professor of Electrical Engineering and Chairman of the Center of Applied Artificial Intelligence, Regional Engineering College, Rourkela, India.

A. K. Pradhan is a Lecturer in the Department of Electrical Engineering, University College of Engineering, Burla, India.

G. Panda is Professor and Head of the Department of Applied Electronics and Instrumentation Engineering, Regional Engineering College, Rourkela, India.