A fast and accurate distance relaying scheme using an efficient radial basis function neural network

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

Regional Engineering College, Rourkela 769008, India

Abstract

The paper presents a new approach for classification and location of faults on a transmission line using a newer version of radial basis function neural network (RBFNN) which provides a more efficient approach for training and computation. The input data to the RBFNN comprise the normalised peak values of the fundamental power system voltage and current waveforms at the relaying location obtained during fault conditions. The extraction of the peak components is carried out using an extended Kalman filter (EKF) suitably modelled to include decaying d.c., third and fifth harmonics along with the fundamental. The fault training patterns required using the efficient version of RBF neural network are much less in comparison to the conventional RBF network and the choice of neurons and the parameters of the network are systematically arrived without resorting to trial and error calculations. The new approach provides a robust classification of different fault types for a variety of power system operating conditions with resistance in the fault path. Further a new fault location strategy is formulated using four neural networks, one each for the major category of faults like LG, LL, LLG and LLL faults. The proper feature selection for the networks results in an accurate and fast distance relaying scheme.

Keywords: Neural network; Extended Kalman filter (EKF); Harmonics

1. Introduction

A digital distance relay essentially measures the impedance up to the point of fault from digitized voltage and current samples at the relaying location. The changing operating conditions of the power system and the resistance in the fault path to ground can introduce errors in the impedance estimation routine and thus the digital relay may either overreach or underreach. It has been recently shown in several publications that the artificial neural network (ANN) can be trained to become a pattern classifier [1–4] for a faulted power transmission system and can classify the different types of fault successfully. The speciality of the ANN based distance protection is that it does not explicitly use the impedance information as the basis of information rather learns from examples presented to it during training. The conventional approach is to use an ANN with instantaneous sampled values of voltage and current waveforms or magnitudes of the post-fault voltage and current phasors of the three phases as inputs and the neurons in the output layer generate a binary value, 0 or 1. The back propagation (BP) algorithm is normally used for training the network and the weights of the network are frozen, once a sufficiently large number of patterns is presented to the network and convergence is reached. Although, BP produces decision surfaces that effectively separate training examples of normal fault classes, this does not result in the most plausible or robust classifier. Further, BP networks have no mechanism to detect when a case to be classified falls in a region with no training data [4]. This is a serious drawback as the power system operates in a wide range of system and fault conditions.

The radial basis function neural network known as radial basis function neural network (RBFNN) has been considered recently for pattern classification in a distance relaying scheme because of its ability to discern faults with data falling outside the training pattern [4]. Although the classical RBFNN scheme produces good

* Corresponding author.

results, the choice of number of hidden units, the number of input sets and the parameters of the network are varied by trial and error. This approach with digitised samples of current and voltage as inputs (24 in number) also results in the use of a large training set and too many hidden units.

This paper, therefore, examines a different architecture for the RBFNN, where a sequential learning scheme [5] and new input parameters are considered, so that a minimal number of hidden units is used. For this scheme, initially no hidden unit is used and more hidden units are added if the pattern presented to the network is different from the earlier one. Further the RBFNN uses a Gaussian function for the computational units, whose centers, spreads and weights of the network are updated using an extended Kalman filter (EKF) based algorithm. A pruning strategy is used to remove the hidden units, which does not result in any change in the network output. Using this new efficient version of RBFNN, the training time and number of hidden neurons are reduced drastically. The classification scheme becomes 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 is used to generate fault data with different system condition, source impedance, fault inception angle and fault resistance values. After the fault is successfully classified, the fault locator neural network is activated. The fault locator block comprises four RBFNNs, one each for the category of faults LG, LL, LLG and LLL. This approach produces accurate location unlike the earlier procedure of using only a single neural network for fault location [2,4]. Several test results are given in the paper to highlight the effectiveness of the new approach.

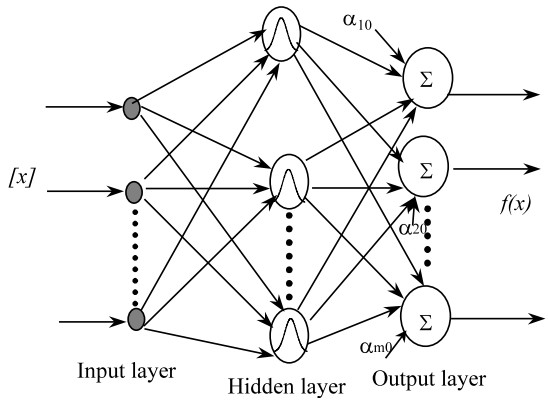

Fig. 1. Radial basis function neural network structure.

2. RBF neural network

The structure of a RBF neural network is shown in Fig. 1. The output of the network is:

$$\hat{y}_n = f(x_n) = \alpha_{m0} + \sum_{k=1}^{K} \alpha_{mk}\phi_k(x_n) \tag{1}$$

For each input x_n, n represents the time index, $K=$ number of hidden units, $\alpha_{mk}=$ connecting weight of the k-th hidden unit to output layer, $\alpha_{m0}=$ bias term, m is the number of output.The value of $\phi_k(x_n)$ is given by:

$$\phi_k(x_n) = \exp(-\frac{1}{\sigma_k^2}\|x_n - \mu_k\|^2) \tag{2}$$

where μ_k is the centre vector for the kth hidden unit and σ_k is the width of the Gaussian function, $\|\ \|$ denotes the Euclidean norm.

The network begins with no hidden units and as observations are received, new hidden units are added by taking some of the input data. The network is trained using the EKF approach. Further, to have a compact network structure a pruning strategy is incorporated. The summary of the algorithm for a single output case is given below.

2.1. Step 1

The error at the nth time step between estimated output $y_n$and desired output y_n is:

$$e_n = |y_n - \hat{y}_n)|$$

Define the rms output error:

$$e_{\mathrm{rms}_n} = \sqrt{\frac{\sum_{i=n-M+1}^{n} [y_i - \hat{y}]^2}{M}} \tag{3}$$

If $e_n > e_{\min}$ and
$\|x_n - \mu_{nr}\| > \epsilon_n$
and $e_{\mathrm{rms}_n} > e'_{\min}$ then,
allocate a new hidden unit with,

$$\alpha(k+1) = e_n,\ \mu_{k+1} = x_n \ \text{ and } \ \sigma_{\kappa+1} = \rho\|x_n - \mu_{nr}\| \tag{4}$$

where$\epsilon_n = \max\{\gamma^n \epsilon_{\max},\ \epsilon_{\min}\}$, $(0<\gamma<1)$, n, the time index $e_{\min}$, $e'_{\min}$, $\epsilon_{\max}$, $\epsilon_{\min}$ are threshold values and ρ is an overlap factor $(0<\rho\le 1)$, μ_{nr} is a center of a hidden unit whose distance from x_n is the nearest among those of all the other hidden unit centers, $M=$ the size of a sliding data window which covers a number of latest observations for calculating the rms output errors e_{rms_n}.

2.2. Step 2

If the above conditions are not fulfilled the parameters of RBFNN are updated by the EKF (modelled for the purpose) as:

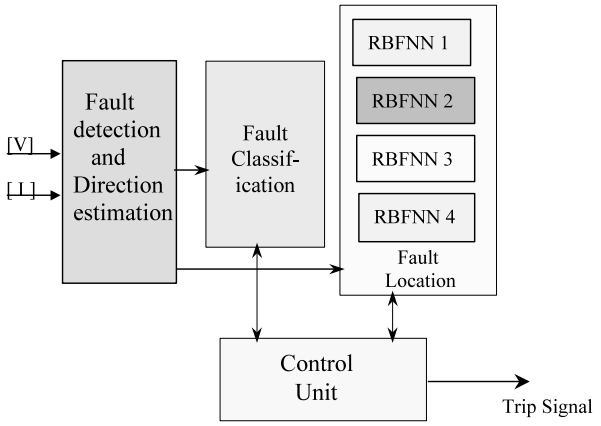

Fig. 2. The distance protection scheme.

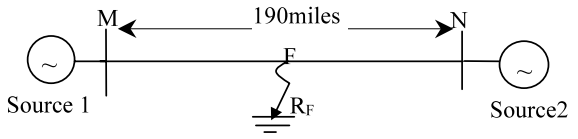

Fig. 3. About 230 kV transmission system.

$$\boldsymbol{W}_n = \boldsymbol{W}_{n-1} + K_n e_n \tag{5}$$

where the parameter vector $\boldsymbol{W}$ for single output case is given by:

$$\boldsymbol{W} = [\alpha_0, \alpha_1, \mu_1^{\mathrm{T}}, \sigma_1, \ldots \alpha_K, \mu_K^{\mathrm{T}}, \sigma_K]^{\mathrm{T}} \tag{6}$$

the Kalman gain K_n is provided in the Appendix A.

2.3. Step 3

- The outputs $O_k^n (k = 1, 2, \ldots K)$ of all hidden units are then computed.
- Find the largest absolute hidden unit output $|O_{\max}^n|$.
- Calculate the normalized value for each hidden unit $r_k^n = |O_k^n / O_{\max}^n| (k = 1, 2, \ldots k)$. If $r_k^n < \delta$ (a threshold value) for M_I consecutive observations, then prune the k-th hidden neuron.

3. Application of RBF neural network to distance protection

The distance protection scheme utilising RBF networks are shown in Fig. 2. The task of the proposed distance protection is to command the trip signal only when a fault occurs in the transmission line in the forward direction which is the direction away from the bus bar and towards the overhead line it protects. The forward direction needs to be identified so that the relay does not respond for faults occurring at the bus or the line behind the relay. To estimate the direction of the fault we followed the approach of phase angle between voltage and current [6]. Once the fault detection block finds a fault in the forward direction it triggers the fault classifier which again activates the fault location unit. The control block derives the decision of trip or no-trip from the output signals of classifier and locator. In this work, an RBF network along with a ground detection unit [7] classifies the fault types first and upon the output of the classifier the control unit fires the suitable RBF network in the locator block. There are four fault locating RBF networks, each representing for a class of fault (LG, LL, LLG and LLL). The inherent benefit of using different networks for four categories of fault is the accuracy of the locator in evaluating the distance of the fault from the relaying point.

For the application of the proposed distance protection strategy based on RBF networks, a 230 kV, 190 mile transmission system as shown in Fig. 3 is considered. To generate data for such a system, digital simulations were performed for different conditions of the system and all shunt faults using EMTDC software package [8]. However, during the first cycle of fault inception the voltage and current signals are badly corrupted by noise, in the form of d.c. offset and frequencies above fundamental. For estimation of the fundamental components from such signals an extended Kalman filter [9] is used in this work. A sampling rate of 1 kHz in a 50 Hz power system is considered for the purpose.

4. Training and testing of RBFNN classifier

The majority of power system protection techniques are involved in defining the system's state through identifying the pattern of the associated voltage and current waveforms. Therefore, a distance protection task can be essentially treated as a pattern recognition/classification problem where neural networks are strong candidates. One of the key issues in neural network designing is the proper selection of features for the particular task. A general approach for this may be to consider a variable as a feature that provides more information for classification of patterns than those not considered. Further pre-processing of signals may drastically reduce the size of the network structure and the global performance of the network may be superior.

Conventional 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. In this work, therefore, the idea of phasors is being integrated with RBF neural network to derive a robust fault classifier for protection of the transmission line covering 80% of the line. Normalised values of post fault peaks of fundamental components of voltages and currents of

the three phases are considered as input vector (6 number) for the network. These peaks are estimated from sampled values of the signal with EKF within half a cycle after fault inception. The RBF 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, a 'ca' line to line fault the output value should be assigned [1 0 1]. The RBF network starts with no hidden units and as the network is exposed to different training patterns, hidden units are added upon novelty of that training set. The training sets are 28 in numbers which include data for 10, 40 and 80% fault location for 0° fault inception angle and at different conditions of the system and seven types of faults (ag, bg, cg, ab, bc, ca, abc). In comparison to earlier approaches [4], the choice of hidden neurons are no more arbitrary, rather a sequential learning and pruning strategy optimally fix the number of such neurons. Furthermore, for updating the parameters of the network while training, only an EKF [9] is used unlike K-means clustering for μ, heuristic approach for σ and multiple regression for weights as in [4]. This clearly demonstrates the simplicity of the strategy in the learning process of the network. The parameters used during training are $\epsilon_{\max} = 0.05$, $\epsilon_{\min} = 0.02$. $\gamma = 0.96$, $e_{\min} = 0.02$ and $e'_{\min} = 0.02$, $\rho = 1$, $P_0 = I$, $Q = 0.05$, $R = 1.0I$, where $I =$ unit matrix of appropriate dimension, $M = 4$, $\delta = 0.0001$ and $M_I = 10$. The final structure of the RBF network becomes 6-22-3. The proper feature selection reduces the number of inputs to six only as compared with 24 as proposed in [4].

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. Tables 1–4 present some of the test results for the faulted transmission line. Table 1 shows the performance of minimal RBFNN 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 reflect the state of involvement of that phase.

Say for 'ab' case with $R_f = 0\ \Omega$ the values 'a' = 1.1257, 'b' = 1.0466 and 'c' = 0.0691 show that the phases associated with the fault are '*a*' and '*b*' only. This classification approach takes a particular phase to be involved with fault if its corresponding value is greater than 0.5 else it categorizes the phase to be 'undisturbed'. For a similar condition as used in Table 1, except the source impedance ratio changed to 90 from 1, Table 2 provides the fault classification results for different faults at 40% of the line. Table 3 shows fault classification for a different condition of load angle (= 20°) at 30° inception angle and at 60% of the line, whereas, Table 4 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 un-

Table 1
Fault at 15% of line at 60° inception angle

Fault type	$R_f = 0\ \Omega$			$R_f = 10\ \Omega$		
	a	*b*	*c*	*a*	*b*	*c*
ag	1.2717	0.2699	−0.0313	1.2783	0.2873	−0.0379
bg	0.3378	0.9995	0.2475	0.3636	0.9676	0.2705
cg	0.2213	0.0176	0.9892	0.2679	0.0197	0.9412
ab	1.1257	1.0466	0.0691	1.0804	1.0522	0.0622
bc	0.0911	1.0051	1.1514	0.0549	1.0153	1.1103
ca	1.1411	0.0873	0.9712	1.1195	0.0720	0.9546
abc	1.1483	1.0524	1.0601	1.1545	1.0730	1.0615

Table 2
Fault at 40% of line at 45° inception angle

Fault type	$R_f = 0\ \Omega$			$R_f = 10\ \Omega$		
	a	*b*	*c*	*a*	*b*	*c*
ag	0.9615	0.2185	0.0041	1.0366	0.4616	−0.0524
bg	0.2271	1.0370	0.4280	0.1920	1.0719	0.4341
cg	0.1352	0.0134	1.14076	0.1490	0.0096	1.14277
ab	0.8618	0.8937	0.1462	0.8618	0.8938	0.01459
bc	0.2441	0.8042	0.7451	0.2432	0.8044	0.7449
ca	0.9288	−0.0416	1.1513	0.9291	−0.0432	1.1516
abc	0.8561	0.9111	0.9757	0.8542	0.9162	0.9757

Table 3
Fault at 60% of line at 30° inception angle

Fault type	$R_f = 0\ \Omega$			$R_f = 10\ \Omega$		
	a	*b*	*c*	*a*	*b*	*c*
ag	0.9650	−0.0051	0.0327	0.9519	−0.0155	0.0296
bg	−0.0232	0.9937	0.0214	−0.0136	0.9785	0.0229
cg	−0.0038	−0.0032	1.0163	0.0226	0.0092	0.9870
ab	0.9938	0.9793	0.0005	1.0203	0.9552	-0.0082
bc	−0.0001	0.9999	1.1006	0.0363	0.9431	0.9634
ca	1.0320	0.0021	1.1016	0.8982	0.0070	0.9790
abc	0.9951	0.9923	0.9861	0.9949	1.1036	1.0082

Table 4
Fault at 80% of line at 90° inception angle

Fault type	$R_f = 0\ \Omega$			$R_f = 10\ \Omega$		
	a	*b*	*c*	*a*	*b*	*c*
ag	0.8926	0.2917	0.2170	0.8719	0.2526	0.1909
bg	0.0820	1.0392	0.0383	0.0433	1.0595	0.0334
cg	−0.2449	0.1078	1.1816	−0.2467	0.1063	1.1599
ab	1.1661	0.8385	0.0785	1.1504	0.8142	0.0746
bc	0.0488	1.0736	0.8837	0.0388	1.1008	0.8788
ca	0.8906	−0.0061	1.1167	0.8671	−0.0366	1.1283
abc	0.9747	0.8871	0.9168	0.9483	0.8221	0.9012
abg	1.1486	0.8434	0.0854	1.1711	0.8407	0.0781
bcg	0.0467	1.0707	0.8839	0.0025	1.1317	0.8636
cag	0.9061	−0.0055	1.1274	0.8720	−0.0356	1.1423

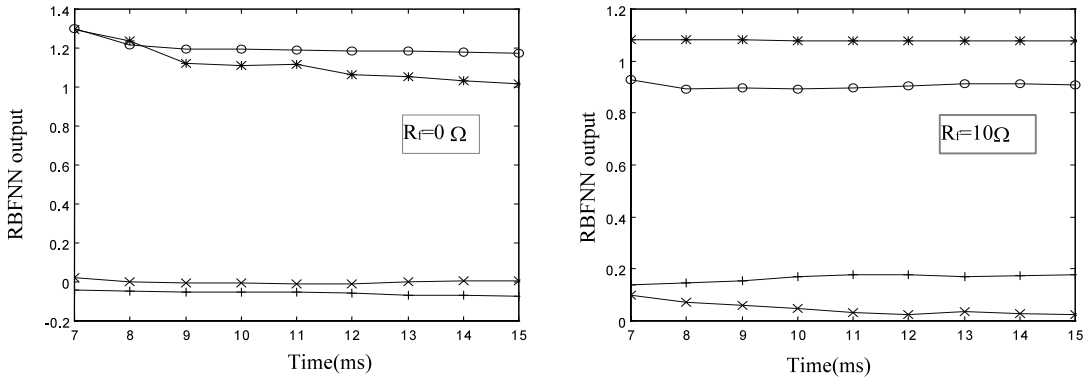

Fig. 4. Convergence loci of the RBFNN for fault at 15% of the line, 'ag' type 60° inception angle phase 'a' —○— 'b' — + — 'c' — × — 'g' —*—.

trained categories of fault; 'abg', 'bcg' and 'cag' etc. which are included in Table 4 (rows 8–10). Observations on all test results ascertain that the RBF network performs excellent even at different inception angle, fault resistance, fault location and prefault loading conditions.

However, another important aspect of the test is to see the consistency in the output of the classifier. Fig. 4 shows the performance of the RBFNN for phase-a to ground fault at 15% of the line for 60° inception angle at $R_f = 0$ and 10 Ω. The classification result shown on a sample-to-sample basis indicates the speed of convergence of the approach. The curves for different phases/ground represent the classifier output during 3/4th of a cycle after fault inception. This result demonstrates that the output of the classifier remains consistent within half a cycle after fault inception and afterward. Similar performances are obtained for other categories of fault. Thus an accurate fault classifier is designed.

Table 5
Index1 values for fault at 15% of the line

Fault type	Index1
ab	0.0038
abg ($R_f = 0\ \Omega$)	0.3969
abg ($R_f = 10\ \Omega$)	0.3840

4.1. Ground detection

Usually it is not possible to identify ground only from peaks of fundamental components of voltages and currents (the input vector to RBFNN). Therefore, the ground detection task is not included in the RBF classifier. In reference [7] for detecting the involvement of ground during fault, a zero sequence current based indicator of the type:

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

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 RBFNN calculations. Test results showing the values of Index1 for '*a*' phase to '*b*' faults at a distance of 15% of the are presented in Table 5.

5. Training and testing of RBFNNs for fault location

As stated earlier, an RBF neural network is proposed for location of each category of fault. Once the fault is classified, the control unit activates the correct fault locating RBF network. The task of RBFNN fault locator is to calculate the normalised distance of the fault point from the relaying point '*M*'. Similar to the input vector in RBFNN classifier, in all the four RBFNN locators the input vector (6 number) consists of normalised peaks of fundamental components of current and voltage as estimated by the EKFs 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. As accuracy is of prime concern these networks require more training sets than the RBFNN classifier. These sets include different fault inception angles, prefault conditions and at different fault distances (10–80%). The total number of such sets is 32 for all four RBFNNs. The networks are trained by the strategy as mentioned in Section 2 with weights being updated by the EKF. The training parameters used are $\in_{\max} = 0.05$, $\in_{\min} = 0.02$, $\gamma = 0.96$, $e_{\min} = 0.02$, and $e'_{\min} = 0.01$, $\rho = 1$, $P_0 = I$, $Q = 0.05$, $R = 1.0I$, where $I =$ unit matrix of appropriate dimension $M = 4$, $\delta = 5\text{e}^{-5}$ and $M_I = 12$. The numbers of hidden units found for the LG, LL, LLG and LLL networks are 23, 25, 26 and 31, respectively. Some of

Table 6
(LG) fault location distance

Distance (%)	15		25		35		45		55		65		75	
R_f (Ω)	0	10	0	10	0	10	0	10	0	10	0	10	0	10
Error (%)	2.77	0.49	0.42	1.87	1.21	2.63	0.68	1.64	2.47	1.82	1.91	1.66	2.13	2.54

Table 7
(LL) fault location distance

Distance (%)	15		25		35		45		55		65		75	
R_f (Ω)	0	10	0	10	0	10	0	10	0	10	0	10	0	10
Error (%)	1.61	1.83	4.11	3.04	1.74	0.04	2.21	2.33	3.42	3.03	4.05	3.40	2.63	2.95

Table 8
(LLG) fault location distance

Distance (%)	15		25		35		45		55		65		75	
R_f (Ω)	0	10	0	10	0	10	0	10	0	10	0	10	0	10
Error (%)	1.25	2.19	2.10	3.33	3.82	3.41	2.77	2.86	2.85	2.87	4.18	4.38	1.18	1.19

Table 9
(LLL) fault location distance

Distance (%)	15		25		35		45		55		65		75	
R_f (Ω)	0	10	0	10	0	10	0	10	0	10	0	10	0	10
Error (%)	2.11	2.67	2.16	1.58	0.46	0.01	0.82	1.41	4.29	4.35	3.38	3.67	2.28	2.57

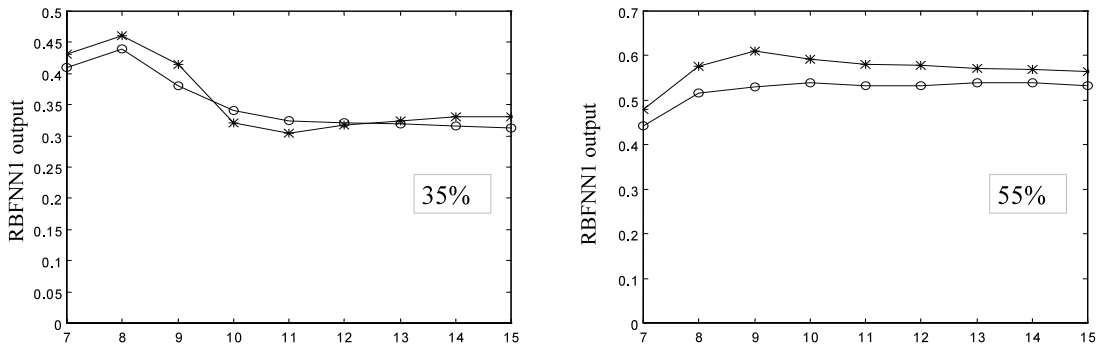

Fig. 5. Convergence loci of RBFNN1 for 'ag' fault, 60° inception angle at a distance of 35 and 55%, R_f, 0 Ω, —○—; and R_f, 10 Ω, —*—.

the test results are presented below where the estimation accuracy of the locators are shown.

Table 6 shows some of the test results for RBFNN locator (LG) at a different condition of the power system, at 60° of inception angle, for both without fault resistance and with 10 Ω of fault resistance cases and at different locations of the fault. Similar results for LL, LLG and LLL RBFNN fault locators are shown in Tables 7–9. 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 was less than 5%. The percentage error was computed as:

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

The RBFNNs calculate the fault distance within 80% of the line with high accuracy and justifies the purpose of using separate fault locating RBFNNs for different fault categories.

Further the convergence speed of the networks is observed by exposing the scheme on a sample to sample basis. Fig. 5 shows the RBFNN1 output for fault cases at 35 and 55% of the line at a fault inception angle of 60° for 'ag' fault type. This figure demonstrates the fault distance estimation results on a sample to sample basis within 3/4th of a cycle after fault inception for different fault resistances. Similar results are obtained for other RBFNN fault locators. It is observed that all the networks calculate the fault distance up to 80% of the line with high accuracy at different prefault operating conditions, inception angles, fault resistances and fault locations.

6. Conclusions

The paper investigates a new high speed distance relaying scheme based on RBFNNs. A single RBF network has been proposed to classify the type of fault, whereas one of the four other RBFNNs is selected to estimate the location of the fault. The training of RBFNNs provides a compact framework in selecting the number of hidden units by employing sequential learning and pruning strategy. The training time for updating the parameters is significantly reduced by using a single EKF algorithm. The trained networks are capable of providing fast and precise classification and location of fault for a variety of system conditions. Furthermore, the proper feature selection in the strategy reduces the number of training data and the size of the RBFNN structures, which is suitable for real time application.

Appendix A

The Kalman gain K_n is:

$$K_n = P_{n-1}a_n[R_n + a_n^{\mathrm{T}}P_{n-1}a_n]^{-1} \tag{A1}$$

where T = transpose of a quantity, R_n = measurement noise variance. The covariance matrix P_n is computed as:

$$P_n = [I - K_n a_n^{\mathrm{T}}]P_{n-1} + QI \tag{A2}$$

where Q is the noise covariance matrix and I is the identity matrix of proper dimension. In EKF principle, the observation vector a_n for single output case is obtained as:

$$a_n = \left.\frac{\partial f(x_n)}{\partial \boldsymbol{W}}\right|_{\boldsymbol{W}=\boldsymbol{W}_{n/n-1}}$$
$$= \left[1, \phi_1(x_n), \phi_1(x_n)\frac{2\alpha_1}{\sigma_1^2}(x_n-\mu_1)^{\mathrm{T}}, \phi_1(x_n)\frac{2\alpha_1}{\sigma_1^3}\|x_n - \mu_1\|^2, \cdots\cdots \phi_k(x_n), \phi_K(x_n)\frac{2\alpha_K}{\sigma_K^2}(x_n-\mu_K)^{\mathrm{T}}, \phi_K(x_n)\frac{2\alpha_K}{\sigma_K^3}\|x_n-\mu_K\|^2\right]^{\mathrm{T}} \quad \text{(A3)}$$

References

[1] T. Dalstein, B. Kulicke, Neural network approach to fault classification for high speed protective relaying, IEEE Trans. Power Deliv. 10 (2) (1995) 1002–1011.

[2] Q.Y. Xuan, Y.H. Song, A.T. Johns, R. Morgan, D. Williams, Performance of an adaptive protection scheme for series compensated EHV transmission systems using neural networks, Electric Power Syst. Res. 36 (1996) 57–66.

[3] D.V. Coury, D.C. Jonge, Artificial neural network approach to distance protection, IEEE Trans. Power Deliv. 13 (1) (1998) 102–108.

[4] Y.H. Song, Q.Y Xuan, 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, (1997), 553–565.

[5] L Yingwei, N. Sundarrajan, P. Saratchandran, Performance evaluation of a sequencial minimal radial basis function (RBF) neural network learning algorithm, IEEE Trans. Neural Networks 9 (2) (1998) 308–318.

[6] S.K. Chakrabarty, C.V. Nayar, N.R. Achuthan, Applying pattern recognition in distance relaying, IEE Proc. C (Part 1 and 2) 139 (4) (1992) 301–314.

[7] M. Akke, J.S. Thorp, Some improvements in the three-phase differential equation algorithm for fast transmission line protection, IEEE Trans. Power Deliv. 13 (1) (1998) 66–72.

[8] Manitoba HVDC Research Center, EMTDC–Electromagnetic Transients Simulation Program, (1988).

[9] T.W. Hilands, S.C.A. Thomopoulos, Nonlinear filtering methods of harmonic retrieval and model order selection in Gaussian and non-Gaussian noise, IEEE Trans. Signal Process. 45 (4) (1997) 983–994.