Transmission line distance relaying using machine intelligence technique

S.R. Samantaray and P.K. Dash

Abstract: A new approach for distance relaying of transmission line using machine intelligence technique such as support vector machine (SVM) is presented. The proposed SVM technique is used for faulty phase selection and ground detection in different fault situations that occur on large power transmission line. Post-fault current and voltage samples for one-fourth cycle (five samples) are used as inputs to SVM 1, which provide output for faulty phase selection. SVM 2 is trained and tested with zero-sequence components of fundamental, third and fifth harmonic components of the post-fault current signal to provides the involvement of ground in the fault process. The polynomial and Gaussian kernel SVMs are designed to provide the most optimised boundary for classification. The total time taken for faulty phase selection and ground detection is 10 ms (half cycle) from the inception of fault. Also the proposed technique is tested on experimental set-up with different fault situations. The test results are compared with those of the radial basis function neural network and were found to be superior with respect to efficiency and speed. The classification test results from SVMs are accurate for simulation model and experimental set-up, and thus provide fast and robust protection scheme for distance relaying in transmission line.

1 Introduction

Different types of transient phenomena occur on the power transmission line. From these transient phenomena, faults on transmission lines need to be detected, classified, located accurately and cleared as fast as possible. Distance relaying techniques based on measurement of impedance at the fundamental frequency between the fault location and the relaying point have received widespread attention. The accuracy of the fault classification and location also depends on amplitude of DC offset and harmonics in comparison to the fundamental component. Fourier transform, differential equation, waveform modelling, Kalman filters and wavelet transform are some of the techniques used for fault detection and location calculation [1–6]. Some of the recent papers in this area [3, 4, 6] have used only the sampled current values at the relaying point during faults to classify fault types and to calculate distance. To obtain more satisfactory results, however, wavelet filters having longer length and more levels of wavelet decomposition must be employed. Consequently, more processing time is required, which is a drawback for protection relays. The Kalman filtering approach has its limitation, as fault resistance cannot be modelled and further it requires a number of different filters to accomplish the task.

The speed and accuracy of distance relays of transmission lines can be improved by accurate and fast faulty phase selection, which is the primary requirement for protective relaying to start and trip correctly. In addition, faulty phase selection can be used to increase system stability by allowing single-pole tripping and autoreclosure. Conventional approaches to phase selection based on power–frequency measurements have disadvantages due to fault resistance, fault distance, influence of mutual coupling from adjacent lines, reactance effect, incomplete knowledge of system parameters and so on. In this regard, some new techniques have been adopted. A method based on initial current travelling waves is presented in [7]. However, these approaches lead to increased hardware requirement. Travelling waves, being high-frequency signals, are difficult to separate from interference noise. In recent years, techniques using artificial neural networks (ANN) and fuzzy logic have been employed in faulty phase selection [5, 8, 9] due to their superior ability to learn and generalise from training patterns. However, in the fault classification and location tasks, the neural networks cannot produce accurate results due to the inaccuracies in the input phasor data and the requirement of a large number of neural networks for different categories of faults. Back propagation neural network, radial basis function neural network (RBFNN) and fuzzy neural network are employed for adaptive protection of such a line where the protection philosophy is viewed as a pattern classification problem. The networks generate the trip or block signals using a data window of voltages and currents at the relaying point. However, the above approaches are sensitive to system frequency changes and require large training sets and training time and a large number of neurons.

This paper presents a new approach for faulty phase selection and ground detection using support vector machine (SVM). An SVM [10–14] is a relatively new machine learning method that optimises model on training data by solving a quadratic program (QP). In essence, an SVM finds the maximal separating hyperplane in feature space. It is computationally efficient because the transformation to feature space need not be done explicitly because

doi:10.1049/iet-gtd:20070160

Paper first received 16th April and in revised form 6th July 2007
S.R. Samantaray is with the National Institute of Technology, Rourkela, Orissa, India

P.K. Dash is with Center for Research in Electrical, Electronics and Computer Engineering, Bhubaneswar, Orissa, India

E-mail: sbh_samant@yahoo.co.in

dot products in feature space can be represented by kernel functions. The SVM-based classification is a modern machine learning method that is rarely used in fault classification even if it has given superior results in various classification and pattern recognition problems such as in text categorisation [15] or phoneme recognition [16]. Currently, there exist only a few publications that concentrate on developing fault diagnostic methods based on SVM techniques [17, 18].

SVM has advantages over traditional approaches such as neural networks for the following reasons.

1. Good generalisation performance – once it is presented with a training set, it is able to learn a rule, which can correctly classify a new object quite often.
2. Computational efficiency – it is efficient in terms of speed and complexity.
3. Robust in high dimensions – in general, dealing with high-dimensional data is difficult for a learning algorithm because of over-fitting. One of the major reasons for attracting much attention is that SVMs are more robust to this over-fitting than other algorithms.

In the proposed research work, the post-fault current and voltage signals after fault inception are retrieved at the relaying end for all phases at a sampling frequency of 1.0 kHz (20 samples per cycle). The proposed scheme works with the consideration that the fault detection has been done. The pre-fault and post-fault boundaries are detected using the fault detector, which uses a short data window (four samples) algorithm [19]. The final indication of the fault is only given when three consecutive comparisons give the difference more than a specified threshold value. After knowing the fault instance, one-fourth cycle data of fault voltage and current signal (five samples each) are used as features to train and test SVM 1 for faulty phase selection. Similarly, SVM 2 is trained and tested with peak of the zero-sequence components (resulted from zero-sequence analyser) of fundamental, third and fifth harmonic components of the post-fault current signal to provide the involvement of ground in the fault process. The polynomial and Gaussian kernel SVMs are trained with features and labels to provide the most optimised boundary for classification. Thus, first one-fourth cycle data are used for fault detection and not included in the fault classification process to avoid the effect of DC offset on fault classification process. The next one-fourth cycle data are used as input to the SVMs for faulty phase selection and ground detection. Thus the total time taken for faulty phase selection and ground detection is 10 ms from the inception of fault (one-half cycle on 20 ms cycle time)

2 Support vector machine

The SVM is firmly grounded in the framework of statistical learning theory, which characterises the properties of learning machines enabling them to generalise well to unseen data. In SVM, original input space is mapped onto a high-dimensional dot-product space called a feature space, and in the feature space, the optimal hyperplane is determined to maximise the generalisation ability of the classifier. Traditional neural network approaches for the empirical data modelling have suffered from difficulties with generalisation, producing models that may over-fit the data. The SVM learning is gaining popularity due to its many attractive features and promising empirical performance. Also, SVM-based classifiers are claimed to have good generalisation properties compared with conventional classifiers, because in training the SVM classifier, the so-called structural misclassification risk is to be minimised, whereas traditional classifiers are usually trained so that the empirical risk is minimised. Structural risk minimisation minimises an upper bound on the expected risk, as opposed to empirical risk minimisation that minimises the error on the training data. It is this difference that equips SVM with a greater ability to generalise, which is the goal in statistical learning. The SVM is compared with the RBFNN in an industrial fault classification task [17], and it has been found to give better generalisation.

Consider that n-dimensional input x_i($i = 1, \ldots, M$, M is the number of samples) belongs to Class I or Class II and associated labels be $y_i = +1$ for Class I and $y_i = -1$ for Class II. For linearly separable data, we can determine a hyperplane $f(x) = 0$ that separates the data

$$f(x) = \boldsymbol{w}^{\mathrm{T}} x + b = \sum_{j=1}^{n} \boldsymbol{w}_j x_j + b = 0 \qquad (1)$$

where $\boldsymbol{w}$ is an n-dimensional vector and b a scalar. $\boldsymbol{w}$ and b determine the position of the separating hyperplane. Function sign($f(x)$) is also called the decision function. A distinctly separating hyperplane satisfies the constraints $f(x_i) \geq 1$ if $y_i = +1$ and results in

$$y_i f(x_i) = y_i(\boldsymbol{w}^{\mathrm{T}} x_i + b) \quad \text{for} \quad i = 1, \ldots, M \qquad (2)$$

The optimal separating hyperplane decides the maximum margin, the maximum distance between the plane and the nearest data. An example of the optimal separating hyperplane of two data sets is presented in Fig. 1. From the geometry, the margin is found to be $\|\boldsymbol{w}\|^{-2}$. Taking into account the noise with slack variables ξ_i and error penalty C, the optimal hyperplane can be found by solving the following convex quadratic optimisation problem

$$\begin{aligned} \min \quad & \frac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_{i=1}^{M} \xi_i \\ \text{s.t.} \quad & y_i(\boldsymbol{w}^{\mathrm{T}} x_i + b) \geq 1 - \xi_i, \quad \text{for } i = 1, \ldots, M \\ & \xi_i \geq 0 \quad \text{for all } i \end{aligned} \qquad (3)$$

where ξ_i is measuring the distance between the margin and the examples x_i lying on the wrong side of the margin. The calculations can be simplified by converting the problem with Kuhn–Tucker conditions into the equivalent

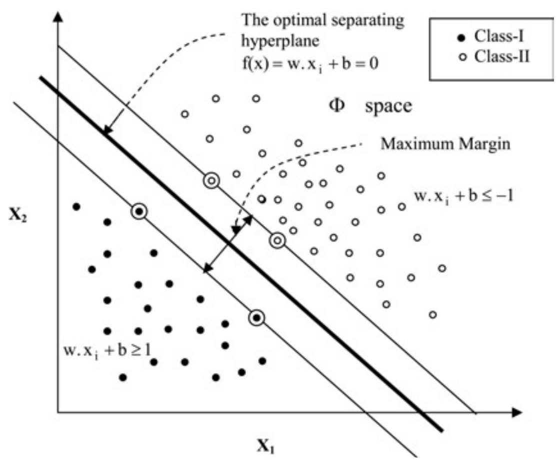

Fig. 1 *f(x) as a separating hyperplane lying in a high-dimensional space*
Support vectors are inside the circles

Lagrange dual problem, which will be

$$\max \quad W(\alpha) = \sum_{i=1}^{M} \alpha_i - \frac{1}{2}\sum_{i,k=0}^{M} \alpha_i \alpha_k y_i y_k x_i^{\mathrm{T}} x_k$$

$$\text{s.t.} \quad \sum_{i=1}^{M} y_i \alpha_i = 0, \quad C \geq \alpha_i \geq 0, \quad i = 1, \ldots, M \tag{4}$$

The number of variables of the dual problem is the number of training data. Let us denote the optimal solution of the dual problem with α^* and w^*. According to the Karush–Kuhn–Tucker condition, the inequality condition in (2) holds for the training input–output (feature and label) pair (x_i, y_i) only if the associated α^* is not 0. In this case, the training example x_i is a support vector (SV). Usually, the number of SVs is considerably lower than the number of training samples making SVM computationally very efficient. The value of the optimal bias b^* is found from the geometry

$$b^* = -\frac{1}{2}\sum_{\mathrm{SVs}} y_i \alpha_i^* (S_1^{\mathrm{T}} x_i + S_2^{\mathrm{T}} x_i) \tag{5}$$

where S_1 and S_2 are arbitrary SVs for Classes I and II, respectively. Only the samples associated with the SVs are summed, because the other elements of optimal Lagrange multiplier α^* are equal to zero.

The final decision function will be given by

$$f(x) = \sum_{\mathrm{SVs}} \alpha_i y_i x_i^{\mathrm{T}} x + b^* \tag{6}$$

Then unknown data example x is classified as follows

$$x \in \begin{cases} \text{Class I} & \text{if } f(x) \geq 0 \\ \text{Class II,} & \text{otherwise} \end{cases} \tag{7}$$

For nonlinear classification problems, SVM with application of kernel function solves the purpose. The input data are mapped onto a high-dimensional feature space, where linear classification is possible. Using a nonlinear vector function $\phi(x) = (\phi_1(x), \ldots, \phi_m(x))$, where $m >> n$, to map the n-dimensional input vector $\boldsymbol{x}$ onto the m-dimensional feature space, the linear decision function in dual form is given by

$$f(x) = \sum_{\mathrm{SVs}} \alpha_i y_i \phi^{\mathrm{T}}(x_i)\phi(x) \tag{8}$$

Working in the high-dimensional feature space enables the expression of complex functions, but it also generates problems. High dimensionality creates the problem of over-fitting and computational problems occur due to the large feature vectors. The over-fitting problem is solved with application of the maximal margin classifier, and kernels provide solution to computational problem.

A function that returns a dot product of the feature space mappings of original data points is called a kernel, $\boldsymbol{K}(x, z) = \phi^{\mathrm{T}}(x)\phi(z)$. Applying a kernel function, the learning in the feature space does not require explicit evaluation of ϕ. Using a kernel function, the decision function will be

$$f(x) = \sum_{\mathrm{SVs}} \alpha_i^* y_i \boldsymbol{K}(x_i x) \tag{9}$$

and the unknown data example is classified as before. The values of $\boldsymbol{K}(x_i, x_j)$ over all training samples $i, j = 1, \ldots, M$, the kernel matrix, which is a central structure in the kernel theory. Mercer's theorem states that any symmetric positive-definite matrix can be regarded as a kernel matrix. In this work, we used the polynomial kernel of the following form

$$\boldsymbol{K}(x, z) = (x^{\mathrm{T}} z + 1)^n \tag{10}$$

where n represents the degree of the inner product kernel. Similarly, the Gaussian kernel used in this study is given in (11), where σ is the width of the Gaussian function

$$\boldsymbol{K}(x, z) = \exp\left\{-\frac{|x - z|^2}{2\sigma^2}\right\} \tag{11}$$

3 System studied

The power system model shown in Fig. 2 is simulated using PSCAD (EMTDC) software package. The relaying point is as shown in Fig.1, where fault voltage and current signal samples are retrieved for different fault conditions. Fig. 3 shows the fault voltage and current signal for three-phase fault. The power system network consists of two areas of 400 kV generation capacities connected by 300 km long transmission line. The transmission line model chosen for the proposed study is of distributed type.

The transmission line parameters are zero-sequence impedance $(Z_{\mathrm{L0}}) = 96.45 + j335.26\ \Omega$, positive sequence impedance $(Z_{\mathrm{L1}}) = 9.78 + j110.23\ \Omega$, source impedances: $Z_{\mathrm{S}} = 6 + j28.5\ \Omega$, $Z_{\mathrm{R}} = 1.2 + j11.5\ \Omega$, source voltages: $E_{\mathrm{S}} = 400$ kV, $E_{\mathrm{R}} = 400\angle\alpha$ kV, where α is the load angle in degrees.

The power system model is simulated at 1.0 KHz sampling frequency. The voltage and current signals for different fault conditions are retrieved at the relaying point and fed to the SVMs for faulty phase selection and ground detection. The proposed SVM-based relaying scheme is shown in Fig. 4.

The proposed algorithm is also tested on a physical transmission line model (experimental set-up). The transmission line consists of two 150 km π sections (total 300 km) and charged with 400 V, 5 kV A synchronous machines at one end and 400 V at the load end. The three-phase voltage and current are stepped down at the relaying end with potential transformer of 400/10 V and current transformer of 15/5 A, respectively. Data are collected using PCL-208 Data Acquisition Card, which uses 12-bit successive approximation technique for A/D (analogue to digital) conversion. The card is installed on a personal computer (P-4) with a driver software routine written in C++. It has six I/O channels with input voltage range of ± 5 V. Data are colleted with a sampling frequency of 1.0 kHz.

4 Computational results

This section deals with the training and testing results obtained from the corresponding SVMs for faulty phase selection and ground detection. Both polynomial and Gaussian kernel-based mappings are used to obtain the

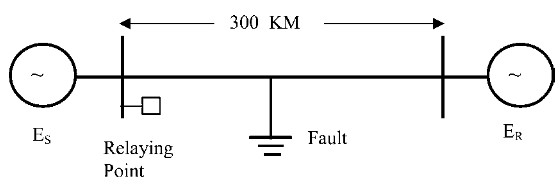

Fig. 2 *Transmission line model*

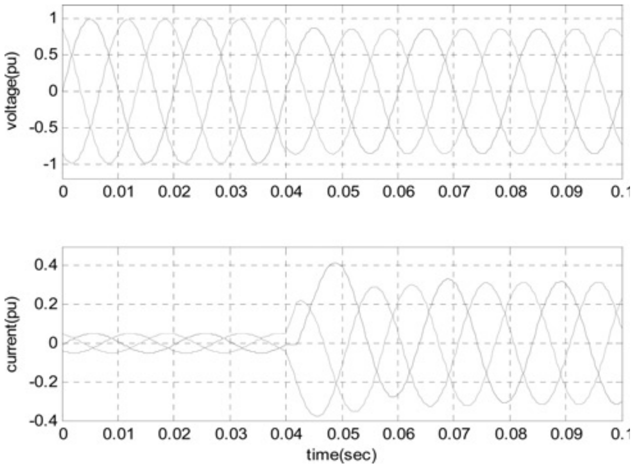

Fig. 3 *Fault voltage and current signal for three-phase fault*

accurate results for classifying faults from unfaulted one. The SVMs are trained with 300 data sets and validated for 300 data sets for each category of fault generated from simulation model and experimental set-up separately. Initially, the RBFNN is tested and the performance is compared with that of the proposed SVM technique for fault classification.

4.1 Fault classification using RBFNN

The fault classification results using RBFNN [20] are given in Table 1. In this approach, a pruning strategy is used to select only a minimal number of hidden neurons by observing their outputs, and if at any stage it is observed that the output of any neuron is insignificant, it is omitted from the hidden layer. The current and voltage samples at the relaying point are retrieved for different fault conditions and peak of the samples are fed to the RBFNN for fault classification. The RBFNN is tested with 300 data sets from simulation model. Table 1 provides the fault classification results using RBFNN for line–ground (L–G), line–line–ground (LL-G), line–line (LL), line–line–line–ground (LLL-G), line–line–line (LLL) faults. The classification rate is 95.12% (maximum) in the case of LLL-G fault and 93.75% for LL fault.

4.2 Fault classification using SVM

4.2.1 Phase selection (SVM 1): Initially, one-fourth cycle fault signal is used for fault detection and the next one-fourth post-fault current and voltage signal samples are used as inputs to the SVM. The corresponding output is either fault or no-fault condition. Five samples of faulted voltage and five samples of faulted current signal from the fault inception are retrieved at the relaying point and the corresponding normalised values are used as input

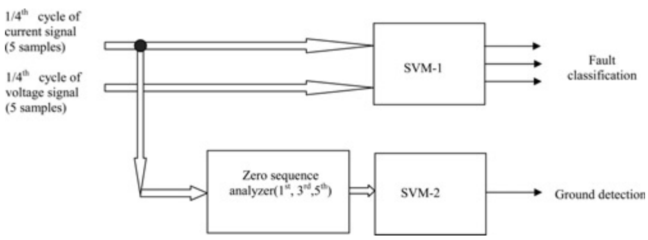

Fig. 4 *Proposed protection scheme*

Fault classification (SVM 1) and Ground detection (SVM 2)

Table 1: Classification rates of RBFNN for faulty phase selection

Fault	RBFNN classification rates, %
L–G	94.89
L–G	94.34
LL	93.75
LLL	94.84
LLL–G	95.12

features space (ten points) to the SVMs, termed as x_i. The corresponding output is y_i, which results in 1 for fault and -1 for no-fault condition. The SVM 1 is trained with 300 data sets and tested with 300 data sets for each category of fault, each set comprising of ten data points for x_i as input and $(1, -1)$ for y_i as the corresponding output.

Faults on the line are simulated with various operating conditions including different incident angles δ, fault resistance R_f (10–200 Ω), source capacities and at various locations for all types of shunt faults. These shunt faults are L–G, L–L, L–L–G, L–L–L and LLL–G. In case of shunt faults, 'a–g', 'b–g' and 'c–g' are categorised under L–G fault and 'ab–g', 'bc–g', 'ca–g' under LL–G. Similarly, 'a–b', 'b–c' and 'c–a' correspond to L–L fault and 'abc–g' corresponds to LLL–G fault. 'abc' is categorised under LLL fault. Thus, there are 11 types of shunt faults that occur on the power transmission line.

Here, n stands for the order of the polynomial and σ for the width of the Gaussian function. The bound on the Lagrangian multipliers C is selected 5.0 and the conditioning parameter for QP method, λ is chosen as 1×10^{-5}. Different values of σ with which the SVM is trained and tested are 1.0 and 1.5. Similarly, the values selected for n are 5 and 6. All the above parameters are selected after cross-validation [21–23]. Different values of σ and n are used to make a comparison study on the classification rate and support vectors generated. When the parameter values of the polynomial and Gaussian kernels are changed, the classification rate and the numbers of support vectors on the optimised marginal plane vary accordingly.

Table 2 provides the classification rates and support vectors during the training of SVMs for faulty phase selection and ground detection. The classification rate for faulty phase selection is 99.26% (maximum) with 14 support vectors and for ground detection is 99.69% (maximum) with 13 support vectors. After the SVMs are trained with the training data sets, the SVMs are tested or validated with test data sets.

Table 3 shows the testing results for faulty phase patterns for various operating conditions. As seen from the table, for 'ab–g'(LL-G) fault at 30%, $\delta = 45^\circ$, $R_f = 50\ \Omega$, the SVM 1 outputs for 'a' and 'b' are 1 but output for 'c' phase is -1 for both polynomial and Gaussian kernels, which depicts that fault occurs on 'a' and 'b' phases. Also for 'bc' fault at 50%, $\delta = 60^\circ$, $R_f = 100\ \Omega$ the output for 'b' and 'c' phases are 1 but the output is -1 for 'a' phase. As seen, the mis-classification occurs for the 'abc–g' (LLL–G) fault at 90%, $\delta = 45^\circ$, $R_f = 150\ \Omega$ with source changed, with Gaussian kernel with $\sigma = 1.0$ resulting output of 'a' phase as -1 instead of 1. Table 4 depicts the classification rates at different fault conditions with polynomial and Gaussian kernels of different parameter values during the testing of SVM. The classification rate is 97.25% (minimum) at LLL–G fault with polynomial kernel with

Table 2: Classification rates and support vectors during the training of SVM 1 and 2 for phase selection and ground detection, respectively

Fault	Kernel	Parameter value	Training results for faulty phase selection (SVM 1)		Training results for ground detection (SVM 2)	
			Classification rates, %	Number of support vectors	Classification rates, %	Number of support vectors
L–G	poly	$n = 5$	98.89	19	99.15	15
	poly	$n = 6$	99.26	14	99.01	13
	Gaussian	$\sigma = 1.0$	98.11	15	99.00	12
	Gaussian	$\sigma = 1.5$	98.34	12	99.27	09
LL–G	poly	$n = 5$	98.45	10	98.99	11
	poly	$n = 6$	98.25	11	98.46	08
	Gaussian	$\sigma = 1.0$	98.99	09	99.49	10
	Gaussian	$\sigma = 1.5$	98.97	08	98.98	06
LL	poly	$n = 5$	98.01	10	99.69	13
	poly	$n = 6$	98.87	09	98.36	11
	Gaussian	$\sigma = 1.0$	98.20	08	98.77	09
	Gaussian	$\sigma = 1.5$	98.04	06	98.02	05
LLL	poly	$n = 5$	98.32	19	99.03	17
	poly	$n = 6$	97.98	14	98.00	14
	Gaussian	$\sigma = 1.0$	98.99	10	98.37	12
	Gaussian	$\sigma = 1.5$	98.03	09	98.53	09
LLL–G	poly	$n = 5$	98.05	12	98.10	10
	poly	$n = 6$	98.89	10	98.98	07
	Gaussian	$\sigma = 1.0$	97.99	08	99.18	09
	Gaussian	$\sigma = 1.5$	98.65	06	99.23	04

$n = 5$ and the classification rate is 98.87% (maximum) for LL–G fault with Gaussian kernel with $\sigma = 1.0$.

4.2.2 Ground detection (SVM 2): The ground detection is done separately by training and testing SVM 2. The peak value of the zero-sequence component of the fundamental third and fifth harmonic components of post-fault current signal are found and are used as the input x_i (three inputs) to the SVM-2 and the corresponding output (y_i) is 1 for the fault involving ground and -1 for fault without involving ground. As the zero-sequence components are pronounced in the case of fault involving ground compared with fault without involving ground, the SVM 2 is trained to provide the classification for ground detection. The SVM 2 is trained and tested with 300 data sets for each category of fault, each set comprising three data points for x_i as input and $(1, -1)$ for y_i as corresponding output.

Table 3 provides the ground detection patterns for different fault conditions. It is found that for 'b–g' (L–G) fault at 10%, $\delta = 30^\circ$, $R_f = 10\ \Omega$, the output is 1, which shows that the fault involves ground. But for 'bc' (L–L) fault at 50%, $\delta = 60^\circ$, $R_f = 100\ \Omega$, the output is -1, which clearly shows that the fault does not involve ground. Also mis-classification is observed for 'abc–g' (LLL–G) fault at 75%, $\delta = 30^\circ$, $R_f = 200\ \Omega$ with source changed, with polynomial kernel for $n = 5$, which produces output -1 instead of 1. Also similar case happens for 'abc–g' (LLL–G) fault at 90%, $\delta = 45^\circ$, $R_f = 150\ \Omega$ with source changed, with Gaussian kernel for $\sigma = 1.0$, which results -1 instead of 1. Table 4 shows the classification rate of the SVM 2 for ground detection. The classification rate is 99.32% (maximum) for LL–G fault with Gaussian kernel with $\sigma = 1.0$ and the 97.89% (minimum) for LL fault with $\sigma = 1.5$. The average classification rate for ground detection for 300 test cases is found to be 98.61% for all types of faults with different operating conditions.

4.2.3 Phase selection and ground detection for experimental data sets: The proposed algorithm is also tested for data from experimental set-up. The trained SVMs are tested with 300 data sets from experimental set-up. The results for faulty phase selection and ground detection are given in Table 5. It is found that the trained SVMs produce accurate results when compared with the results using data from simulation model. The average faulty phase selection rate is 97.96% and the average ground detection is 98.50% compared with faulty phase selection and ground detection rates of 98.02 and 98.62%, respectively, using data from simulation study during testing.

4.3 Discussion

It is found that SVMs 1 and 2 combined together provide accurate results for phase selection and ground detection, respectively. The performance comparison between RBFNN and SVM is given in Table 6. It is observed that when training size reduces, the classification accuracy reduces drastically in the case of RBFNN compared with SVM. For L–G fault, the classification accuracy of RBFNN and SVM is 94.89 and 98.23%, respectively, with 300 training and testing data sets. When the training set decreases to 100 with testing set remaining 300, the classification accuracy of RBFNN and SVM is 87.78 and 96.99%, respectively. Similar observations are made for other fault situations with different training and testing data sets. From the above results, it can be concluded that

Table 3: Testing of SVM 1 for faulty phase patterns and SVM 2 for ground detection patterns

Fault	Kernel	Parameter value	SVM 1 a-phase	SVM 1 b-phase	SVM 1 c-phase	SVM 2 Ground
b–g fault at 10%, $\delta = 30^\circ$, $R_f = 10\ \Omega$	poly	$n = 5$	−1	1	−1	1
	poly	$n = 6$	−1	1	−1	1
	Gaussian	$\sigma = 1.0$	−1	1	−1	1
	Gaussian	$\sigma = 1.5$	−1	1	−1	1
ab–g fault at 30%, $\delta = 45^\circ$, $R_f = 50\ \Omega$	poly	$n = 5$	1	1	−1	1
	poly	$n = 6$	1	1	−1	1
	Gaussian	$\sigma = 1.0$	1	1	−1	1
	Gaussian	$\sigma = 1.5$	1	1	−1	1
‘bc’ fault at 50%, $\delta = 60^\circ$, $R_f = 100\ \Omega$	poly	$n = 5$	−1	1	1	−1
	poly	$n = 6$	−1	1	1	−1
	Gaussian	$\sigma = 1.0$	−1	1	1	−1
	Gaussian	$\sigma = 1.5$	−1	1	1	−1
‘abc’ fault at 70%, $\delta = 45^\circ$, $R_f = 150\ \Omega$	poly	$n = 5$	1	1	−1	−1
	poly	$n = 6$	1	1	1	−1
	Gaussian	$\sigma = 1.0$	1	1	1	−1
	Gaussian	$\sigma = 1.5$	1	1	1	−1
‘abc–g’ fault at 90%, $\delta = 45^\circ$, $R_f = 150\ \Omega$ with source changed	poly	$n = 5$	1	1	1	1
	poly	$n = 6$	1	1	1	1
	Gaussian	$\sigma = 1.0$	−1	1	1	−1
	Gaussian	$\sigma = 1.5$	1	1	1	1
‘ca–g’ fault at 45%, $\delta = 60^\circ$ $R_f = 100\ \Omega$	poly	$n = 5$	1	−1	1	1
	poly	$n = 6$	1	−1	1	1
	Gaussian	$\sigma = 1.0$	1	−1	1	1
	Gaussian	$\sigma = 1.5$	1	−1	1	1
‘c–g’ fault at 85%, $\delta = 60^\circ$, $R_f = 150\ \Omega$	poly	$n = 5$	−1	−1	1	1
	poly	$n = 6$	−1	−1	1	1
	Gaussian	$\sigma = 1.0$	−1	−1	1	1
	Gaussian	$\sigma = 1.5$	−1	−1	1	1
‘ab’ fault at 95%, $\delta = 45^\circ$, $R_f = 200\ \Omega$	poly	$n = 5$	1	1	−1	−1
	poly	$n = 6$	1	1	−1	−1
	Gaussian	$\sigma = 1.0$	1	1	−1	−1
	Gaussian	$\sigma = 1.5$	1	1	−1	−1
‘abc–g’ fault at 75%, $\delta = 30^\circ$, $R_f = 200\ \Omega$ with source changed	poly	$n = 5$	1	−1	1	−1
	poly	$n = 6$	1	1	1	1
	Gaussian	$\sigma = 1.0$	1	1	1	1
	Gaussian	$\sigma = 1.5$	1	1	1	1

SVM is the better approach to learn small size of data patterns compared with RBFNN, which reflects the computational efficiency of designed SVM classifier.

The fault classification results obtained are accurate for data from simulation model and experimental set-up. In the proposed algorithm, the first one-fourth cycle data after fault inception are used for fault detection and the next one-fourth cycle data are used as input to the SVMs for phase selection and ground detection, and the total time taken in the phase selection and ground detection is 10 ms from the inception of fault (one-half cycle on 20 ms cycle time). Thus, the proposed SVM protection scheme provides a fast and robust fault classifier for distance relaying. While comparing the efficiency and speed of the algorithms, the RBFNN suffers because of complex structure and high computation time. Because of higher computational efficiency and speed, the proposed SVM technique can be extended to develop the real-time protection relays for large power system network. The algorithm is simple to develop and provide accurate results for different fault situations with wide variations in operating conditions of the power system network.

The performance of the designed SVM classifier for fault classification in complex power networks such as looped system and distributed power networks is studied. In the case of above networks, fault current contains high decaying DC offset and harmonics compared with the fundamental component. Thus, the designed SVMs may fail to provide accurate fault classification. As the SVM parameters are selected using a cross-validation process, it does not provide accurate SVM parameters for wide variations in input sets, resulting in inaccurate classification. In this case, selection of cross-validation process has a major role to play, and the input data are to be tested on different cross-validation process to provide SVM parameters for accurate fault classification. Thus, the above

Table 4: Classification rates of SVM 1 for faulty phase selection and SVM 2 for ground detection during testing

Fault	Kernel	Parameter value	Classification rates for phase selection (SVM 1), %	Classification rates for ground detection (SVM 2), %
L–G	poly	$n = 5$	98.16	99.05
	poly	$n = 6$	98.19	98.74
	Gaussian	$\sigma = 1.0$	98.23	98.98
	Gaussian	$\sigma = 1.5$	97.89	99.14
LL–G	poly	$n = 5$	98.15	98.96
	poly	$n = 6$	98.09	98.25
	Gaussian	$\sigma = 1.0$	98.87	99.32
	Gaussian	$\sigma = 1.5$	97.84	98.39
LL	poly	$n = 5$	97.84	99.25
	poly	$n = 6$	98.29	98.21
	Gaussian	$\sigma = 1.0$	97.99	98.69
	Gaussian	$\sigma = 1.5$	97.87	97.89
LLL	poly	$n = 5$	98.01	98.47
	poly	$n = 6$	97.56	97.98
	Gaussian	$\sigma = 1.0$	98.68	98.02
	Gaussian	$\sigma = 1.5$	97.87	98.07
LLL–G	poly	$n = 5$	97.25	97.99
	poly	$n = 6$	98.69	98.77
	Gaussian	$\sigma = 1.0$	97.86	99.05
	Gaussian	$\sigma = 1.5$	98.15	99.06

Table 5: Classification rates of SVM 1 for faulty phase selection and SVM 2 for ground detection for experimental data sets

Fault	Kernel	Parameter value	Testing results for faulty phase selection SVM 1 classification rates, %	Testing results for ground detection SVM 2 classification rates, %
L–G	poly	$n = 5$	97.88	99.01
	poly	$n = 6$	97.11	98.56
	Gaussian	$\sigma = 1.0$	96.98	98.76
	Gaussian	$\sigma = 1.5$	98.05	99.02
LL–G	poly	$n = 5$	97.95	98.74
	poly	$n = 6$	97.23	98.11
	Gaussian	$\sigma = 1.0$	98.23	99.25
	Gaussian	$\sigma = 1.5$	97.21	98.27
LL	poly	$n = 5$	98.98	99.32
	poly	$n = 6$	97.98	98.12
	gaussian	$\sigma = 1.0$	98.43	98.56
	gaussian	$\sigma = 1.5$	97.45	97.68
LLL	poly	$n = 5$	98.65	98.32
	poly	$n = 6$	97.12	97.78
	Gaussian	$\sigma = 1.0$	97.99	97.99
	Gaussian	$\sigma = 1.5$	97.76	98.01
LLL–G	poly	$n = 5$	97.65	97.87
	poly	$n = 6$	98.67	98.69
	Gaussian	$\sigma = 1.0$	99.01	98.97
	Gaussian	$\sigma = 1.5$	98.92	98.99

Table 6: **Performance comparison between SVM and RBFNN**

Case	Train/test	Data sets	Fault	Classification test result, % RBFNN	SVM ($\sigma = 1.0$)
Case 1	training	300	L–G	94.89	98.23
	testing	300			
Case 2	training	200		91.21	97.58
	testing	300			
Case 3	training	100		87.78	96.99
	testing	300			
Case 1	training	300	LL–G	94.34	98.87
	testing	300			
Case 2	training	200		91.83	97.69
	testing	300			
Case 3	training	100		87.95	96.01
	testing	300			
Case 1	training	300	L–L	93.75	97.99
	testing	300			
Case 2	training	200		91.87	96.42
	testing	300			
Case 3	training	100		88.12	95.68
	testing	300			
Case 1	training	300	LLL	94.84	98.68
	testing	300			
Case 2	training	200		91.45	97.95
	testing	300			
Case 3	training	100		87.87	96.23
	testing	300			
Case 1	training	300	LLL–G	95.12	97.86
	testing	300			
Case 2	training	200		92.64	96.84
	testing	300			
Case 3	training	100		88.25	95.25
	testing	300			

critical conditions of the power system network and their impacts on SVM classifier are being studied and will be reported later.

5 Conclusions

An SVM based protection scheme (fault classifier) for large power transmission line is presented in this paper. In the proposed technique, one-fourth cycle post-fault current and voltage samples are collected at the relaying point and fed to the SVMs as inputs, and it provides information about the faulty phase and ground involved in the fault process. SVM 1 is trained and tested with the faulted voltage and current samples to provide faulty phase selection, and SVM 2 is trained and tested with the peak of the zero-sequence currents to provide the involvement of ground in the fault process. The polynomial and Gaussian kernels-based SVMs provide faulty phase selection and ground detection with error $<2\%$. The results are compared with the RBF neural networks (previous work) and found better with respect to the efficiency and speed. The proposed method detects and classifies the faults within one-half cycle from the inception of fault (10 ms). Also, the algorithm is tested for experimental set-up and provides accurate results for faulty phase selection and ground detection. Hence, the proposed technique is very fast, accurate and robust to protect large power transmission networks.

6 References

1 Youssef, O.A.S.: 'New algorithms for phase selection based on wavelet transforms', *IEEE Trans. Power Deliv.*, 2002, **17**, pp. 908–914

2 Youssef, O.A.S.: 'Online applications of wavelet transform to power system relaying', *IEEE Trans. Power Deliv.*, 2003, **18**, (4), pp. 1158–1165

3 Youssef, O.A.S.: 'Combined fuzzy-logic wavelet-based fault classification for power system relaying', *IEEE Trans. Power Deliv.*, 2004, **19**, (2), pp. 582–589

4 Osman, A.H., and Malik, O.P.: 'Transmission line protection based on wavelet transform', *IEEE Trans. Power Deliv.*, 2004, **19**, (2), pp. 515–523

5 Martin, F., and Aguado, J.A.: 'Wavelet-based ANN approach for transmission line protection', *IEEE Trans. Power Deliv.*, 2003, **18**, (4), pp. 1572–1574

6 Chanda, D., Kishore, N.K., and Sinha, A.K.: 'A wavelet multi-resolution analysis for location of faults on transmission lines', *Elect. Power Energy Syst.*, 2003, **25**, pp. 59–69

7 Boolen, M.H.J.: 'Traveling wave-based protection of double circuit lines', *Proc. Inst. Elect. Engg. C*, 1993, **140**, (1), pp. 37–47

8 Dalstein, T., and Kluicke, B.: 'Neural network approach to fault classification for high speed protection relaying', *IEEE Trans. Power Deliv*, 1995, **10**, (2), pp. 1002–1011
9 Dash, P.K., Pradhan, A.K., and Panda, G.: 'A novel fuzzy neural network-based distance relaying scheme', *IEEE Trans. Power Deliv.*, 2000, **15**, (3), pp. 902–907
10 Burges, C.J.C.: 'A tutorial on support vector machines for pattern recognition', *Data Min. Knowl. Discov.*, 1998, **2**, (2), pp. 121–167
11 Chapelle, O., and Vapnik, V.N.: 'Model selection for support vector machines' in Solla, S., Leen, T.K., and Muller, K.-R. (Eds.): 'Advances in neural information processing systems' (MIT Press, Cambridge, MA, 2000), vol. 12, pp. 230–236
12 Collobert, R., and Bangio, S.: 'SVMTorch: a support vector machine for large-scale regression and classifcation problems', *J. Mach. Learn. Res.*, 2001, **1**, pp. 143–160
13 Cristianini, N., and Shawe-Taylor, J.: 'An introduction to support vector machines and other kernel-based learning methods' (Cambridge University Press, Cambridge, 2000)
14 Vapnik, V.N.: 'The nature of statistical learning theory' (Springer, New York, 1999)
15 Joachims, T.: 'Text categorization with support vector machines: learning with many relevant features'. Technical Report, University of Dortmund, 1997
16 Salomon, J.: 'Support vector machines for phoneme classification'. MSc Thesis, University of Edinburgh, 2001
17 Ribeiro, B.: 'Support vector machines and RBF neural networks for fault detection and diagnosis'. Proc. 8th Int. Conf. Neural Information Processing, paper 191, 2001
18 Salat, R., and Osowski, S.: 'Accurate fault location in the power transmission line using support vector machine approach', *IEEE Trans. Power Deliv.*, 2004, **19**, (2), pp. 979–986
19 Mann, B.J., and Morrison, I.F.: 'Relaying a three phase transmission line with a digital computer', *IEEE Trans. Power Appar. Syst.,1971*, **90**, (2), pp. 742–750
20 Dash, P.K., and Samantaray, S.R.: 'An accurate fault classification algorithm using a minimal radial basis function neural network', *Int. J. Eng. Intelli. Syst.*, 2004, **12**, (4), pp. 205–210
21 Cawley, G.C., and Talbot, N.L.C.: 'Fast exact leave-one-out cross-validation of sparse least-squares support vector machines', *Neural Netw.*, 2004, **17**, pp. 1467–1475
22 Joachims, T.: 'Estimating the generalization performance of a SVM efficiently'. Proc. 17th Int. Conf. Machine Learning, San Francisco, CA, Morgan Kaufman, 2000
23 Kearns, M., and Ron, D.: 'Algorithmic stability and sanity-check bounds for leave-one-out cross validation'. Proc. 10th Conf. Comput. Learning Theory ACM, New York, 1997 pp. 152–162