FAULT CLASSIFICATION AND LOCATION IN SERIES COMPENSATED TRANSMISSION LINES

A THESIS

Submitted in partial fulfilment of the
requirements for the award of the degree
of
DOCTOR OF PHILOSOPHY
in
ELECTRICAL ENGINEERING

by

PUSHKAR TRIPATHI

DEPARTMENT OF ELECTRICAL ENGINEERING
INDIAN INSTITUTE OF TECHNOLOGY ROORKEE
ROORKEE – 247667 (INDIA)

DECEMBER 2015

INDIAN INSTITUTE OF TECHNOLOGY ROORKEE
ROORKEE

CANDIDATE'S DECLARATION

I hereby certify that the work which is being presented in this thesis entitled **FAULT CLASSIFICATION AND LOCATION IN SERIES COMPENSATED TRANSMISSION LINES** in partial fulfilment of the requirements for the award of the Degree of Doctor of Philosophy and submitted in the Department of Electrical Engineering of the Indian Institute of Technology Roorkee is an authentic record of my own work carried out during a period from July 2009 to December 2015 under the supervision of **Dr. G. N. Pillai**, Associate Professor and **Dr. H. O. Gupta**, Professor, Department of Electrical Engineering, Indian Institute of Technology Roorkee.

The matter presented in this thesis has not been submitted by me for the award of any other degree of this or any other Institute.

(**PUSHKAR TRIPATHI**)

This is to certify that the above statement made by the candidate is correct to the best of our knowledge.

(G. N. Pillai)
Supervisor

(H. O. Gupta)
Supervisor

Date: _____

The Ph.D. Viva-Voce Examination of **Mr PUSHKAR TRIPATHI**, Research Scholar, has been held on ……..

Chairman, SRC

Signature of External Examiner

This is to certify that student has made all the correction in the thesis.

Signature of Supervisors

Head of the Department

ABSTRACT

Series compensation is increasingly identified as an important part of power system, which helps in full utilization of existing transmission infrastructure by increasing the power transfer capability of existing lines. Moreover, variable series compensation has additional advantages such as improvement of stability of system by damping the power swings. However, advantages of series compensation pose extra challenges for the protection of compensated line. Fixed compensation by capacitor has conventional problems such as nonlinear operation of protective equipment of capacitor, current reversal, and voltage reversal. These problems are further aggravated if compensation is provided by thyristor controlled series capacitor (TCSC), because of different operating modes and variable compensation of TCSC.

Fault classification, section identification, and location of fault are few of the protection tasks that become difficult for series compensated line due to above discussed problems. To perform these protection tasks efficiently, this thesis presents new methods using four popular single-hidden layer feedforward networks (SLFNs), namely Support Vector Machine (SVM), Relevance Vector Machine (RVM), Extreme Learning Machine (ELM), and Kernel Extreme Learning Machine (KELM). These SLFNs are chosen for the task as they are getting popularity due to their ability to map high degree of nonlinearity, which is well suited for highly nonlinear nature of the problems addressed.

Performances of the four SLFNs are demonstrated on a digitally simulated two-area system and a 12-Bus system. Although, models of the four SLFNs are same in architecture, their performance based on accuracy, training time, testing time, and number of nodes in hidden layer are different. Therefore, depending on these performance criteria different SLFNs are suitable for different applications e.g. real time or non-real time. While task of fault location is non-real-time, fault classification and section identification can be real-time or non-real-time depending on application. This thesis suggests the suitability of the SLFNs for different tasks.

Similar to any learning machine, for training, cross-validation, and testing of an SLFN, large amount of data is required. Therefore, repeated simulation of the two-area system and 12-Bus system were simulated by considering wide variation in system and fault condition. To make this procedure of data generation fast and convenient, thesis proposes two methods of enhancing batch mode capabilities of an existing electromagnetic transient simulation program. Effectiveness of these two methods is illustrated in two cases, in one case nearly two person-weeks and in another case more than one person-moth were saved.

SVM and RVM have obtained very high accuracy (more than 99.50%) for fault classification and sparse nature of RVM deemed it suitable for real-time application of fault classification. SVM, RVM, and KELM have achieved very high accuracy (more than 99%) for

fault classification and faulty section identification, their appropriateness for applications is discussed in detail.

Furthermore, another new approach of modifying arrangement of input samples to KELM is proposed for improving accuracy of fault location. Although, KELM's testing time is comparatively large, fault location accuracy of KELM is the highest and the time required for training and tuning-parameters is small. Therefore, KELM is found to be most suitable for offline application of fault location. Effect of variation of different parameters on KELM's fault location accuracy is also analyzed.

ACKNOWLEDGEMENTS

First and foremost, I would like to express my deep sense of gratitude to my research supervisors *Dr. G. N. Pillai*, Professor, Department of Electrical Engineering, IIT Roorkee,, and *Dr. H. O. Gupta*, Professor and Director, Jaypee Institute of Information Technology-Sector 128, Noida for their guidance and support to complete the task of this magnitude. Words are inadequate to express their persistent encouragement, painstaking supervision, innovative suggestions, and invaluable help, fructifying this study to its final shape. Their company provided a great opportunity to learn not only in the research area but also in other facets of life. *Dr. Pillai* as a beacon of knowledge always recharged me through his constructive criticisms and thought provoking discussions. His balanced approach towards life and great forbearance has always brought enlightenment in the moments of despair. *Dr. Gupta* has a great capability to dwarf any gigantic problem with his ever-ready practical and witty solutions. His systematic lifestyle has set a benchmark, which I would like to achieve in my life.

I would like to thank *late Dr. J. D. Sharma* for his support and guidance for initial three years of course. May his soul rest in peace.

I would like to acknowledge *Dr. S. P. Srivastava*, Professor and head, Electrical Engineering Department, Indian Institute of Technology Roorkee, for providing me all the infrastructural facilities to carry out this work in the department.

I would like to thank *Dr. B. Das*, Professor (Chairman SRC and Internal expert, SRC), Electrical Engineering Department and *Dr. R. C. Mittal*, Professor (External expert, SRC), Mathematics Department, I.I.T. Roorkee, for giving their invaluable suggestions during course work and guidance in successful completion of this work.

I would like to thanks, Ministry of Human Resource Development (MHRD), Government of India, for providing me financial assistance for carrying out my Ph.D. work.

Next, I wish to convey my deep appreciation to my *fellow research scholars* whose company itself has been a great pleasure and a real help. I would like to thank Mr. Novalio Daratha, Dr. Abhishek Sharma, Dr. Amit Tripathi, Dr. Gaurav Kaushal, Dr. Javed Dhillon, Mr. Sajan, Mr. Amit Garg and all the fellows who helped me directly or indirectly during the entire period of this work.

I avail the privilege to pour on paper, my regards to my *parents***,** Mr. Suraj Prasad Tripathi and Mrs. Shiv Devi. It was their invaluable support only that I could bravely manage to complete this study. My thanks also go to my elder brothers, Mr. Rakesh Tripathi and Mr. Pankaj Tripathi and my sisters Mrs. Rajani Awasthi, Mrs. Kanti Pandey and Mrs. Malti Shukla for their kind blessings, love and care.

No amount of thanks is enough, finally, for my beloved wife Sneh who held the key to my success. Her patience and encouragement kept my motivation up.

I am at loss of words to express my gratitude for all the spiritual gurus and saints for their direct and indirect presence and guidance.

Words seem vocal to express my gratitude but they are dumb to express the grace of the God. As the same electricity manifests itself through different instrument, O Lord thy grace manifests itself through everyone and everything. This famous Sanskrit hymn says it all.

कायेन वाचा मनसेन्द्रियैर्वा बुद्ध्यात्मना वा प्रकृतेः स्वभावात् ।

करोमि यद्यत्सकलं परस्मै सदाशिवायेति समर्पयामि ॥

Whatever I do with my mind, body, speech, other senses of body,
intellect, or with my innate natural tendencies,
I dedicate everything to the ever-auspicious supreme Lord.

(Pushkar Tripathi)

CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1: INTRODUCTION

This chapter introduces the research work. It starts with broader overview of the advantages and challenges posed by series compensated transmission lines. Then, chapter gives a brief survey of literature for fault classification, section identification, and location. The chapter also gives overview of single-hidden layer feedforward neural networks (SLFNs), which were used in the work. Next, scope of work, author's contribution, and thesis outlines are explained.

1.1 GENERAL OVERVIEW

With the growth of a country, the demand for electric power increases rapidly. Increasing demand has led to increase in generation capacity. Generation capacity is being increased either by commissioning new power plants or by enhancing installed capacity of the existing power plants. Since normally generating plants are located in remote locations from load centers, evacuation of large power is required from generating station to load centers. Power can be evacuated either by erecting new transmission lines or by increasing power transfer from existing transmission lines. However, installing new transmission line is capital intensive. Moreover, sometimes right of way for installing new line is not available due to geographical or socio-political reasons. Traditionally, existing transmission lines operate well below their thermal stability limit. Therefore, additional power can be transferred by increasing power transfer through existing transmission lines. This can be achieved by series compensation of transmission line either by installing fixed capacitors (FCs) or a thyristor controlled series capacitors (TCSCs) [1, 2].

1.2 SERIES COMPENSATION

Series compensation can be achieved by cancelling out some of the inductive reactance of transmission line by installing a series capacitor. This can be shown by standard power transfer equation for a transmission line as:

$$P = \frac{E_1 E_2 \sin\delta}{X_L - X_C} \tag{1.1}$$

where P is the power transferred, δ is the phase difference between the voltages of two ends, E_1 and E_2 are the voltage magnitudes of terminal buses of the compensated line, X_L is the inductive reactance of the line, and X_C is the capacitive reactance of compensating device. Power transfer can be increased by increasing capacitive reactance (X_C) of compensating device, while keeping other factors (E_1, E_2, X_L, and δ) constant. Similarly, for same power transfer if bus voltages (E_1 and E_2) are kept constant then for same transmission line (i.e. X_L is also constant) load angle (δ) between buses decreases, thereby steady state margin increases.

Therefore, series compensation by fixed capacitors can increase base power i.e. loading level of the transmission line can be increased near to its thermal stability limits.

Moreover, fixed capacitor can relieve parallel paths from congestion [3]. However, compensation provided by fixed capacitor remains fixed, and outage of a line from its parallel path can violate thermal limits of the compensated line [2]. Moreover, fixed capacitors make shafts of large generators susceptible to damage by increasing the chances of subsynchronous resonance [4]. Additionally, dc-offset-voltage appears during insertion of fixed capacitor.

These limitations of fixed capacitor can be avoided by TCSC, as TCSC can provide fast and continuously controllable compensation. TCSC can change its operation quickly from capacitive to inductive mode and vise-versa as required. Moreover, TCSCs have many other advantages such as improving voltage stability, achieving better transient stability margins than uncompensated line, minimizing losses in overall system by controlling power flow in one line, nullifying dc-offset voltage. Furthermore, during the fault rapid removal and after the fault rapid insertion of the capacitor is possible by controlling the firing pulses of TCSC. Importantly, stability of power system can be improved with the help of TCSC by damping the power swings [5-7]. Moreover, TCSC can efficiently damp out oscillations caused by subsynchronous resonance [8].

Although TCSCs are more beneficial than fixed capacitors, for same level of compensation TCSCs are more expensive than fixed capacitors. Therefore, combination of fixed capacitors and TCSCs are installed as a compensating bank to take advantages of both.

1.3 CHALLENGES POSED BY SERIES COMPENSATION

Despite these advantages, series compensation poses many challenges for protection of transmission line. Protection of series compensation becomes difficult mainly due to the following reasons [9-13]:

- Normally, impedance seen by the relay increases with the increase in fault distance from the relay. However, when capacitor is installed between the fault location and the relay, there is a sudden decrease in impedance seen at the point of capacitor installation. Therefore, conventional impedance relays can overreach if overall impedance seen by the relay is inductive and it can perceive the fault in the reverse direction if the fault is just after capacitor and overall impedance seen by the relay is capacitive.
- During short circuit faults, high level of current can cause dangerous overvoltage. Therefore, to protect capacitor from over voltage protective equipment, such as Metal Oxide Varistor (MOV), triggered air gap, and bypass circuit breakers are installed. During high current faults, one of this equipment may bypass the series capacitor partially or fully. Settings of the relay should be adjusted according to whether the capacitor is conducting fault current or capacitor is bypassed. Moreover, these

protective equipment can act unsymmetrically for different phases. Highly unpredictable nature of operation of the protective equipment makes it further difficult to design a protection algorithm.

- Situation becomes much more complicated when combination of fixed capacitors and TCSCs are used as compensating bank. During fault depending on fault level, one of the capacitor (FC or TCSC) in compensating bank may be bypassed, while other capacitor may remain in circuit, because MOV of one capacitor can have different rating than MOV of other capacitor.
- Phasor estimation during fault becomes difficult due to transients introduced by series capacitor. Therefore, accuracy of protection algorithms, which depend upon calculation of fundamental phasor, may be compromised. If the capacitive reactance of compensating bank is more than the inductive reactance of system then higher frequency transients can appear in fault signals. When the capacitive reactance of the compensating bank is less than the inductive reactance of the system then transients having frequency lower than the system can modulate the fault signal.
- If reactance seen up to faults is capacitive, the voltage seen by relay will be 180 degree out of phase to voltage of uncompensated line. Normally this happens when a fault is near to capacitor installed. Generally, distance algorithms are intended for inductive system. Therefore, protection algorithms will require a revision in this case.
- If the fault is located in such a way that the impedance of the system on one side of the fault is inductive and other side the system is capacitive, then phase of current on capacitive side changes by 180 degree. Therefore, the fault internal to protected transmission line appears to be an external fault.
- The challenges mentioned above are faced by fixed capacitor as well as TCSC. However, TCSC faces additional problems due to different operating modes of TCSC. Generally, TCSC operates in one of the four operating modes, namely Capacitive-vernier mode, Inductive-vernier mode, Bypassed-Thyristor mode, and Blocked-Thyristor mode. If the operating mode of TCSC is not known to the relay then the relay may overreach or underreach depending on the operating mode.

Due to the above challenges posed by series compensation for protection of series compensated line, the conventional protection schemes seem inadequate. Therefore, many protection approaches are explored in literature.

1.4 LITERATURE SURVEY

Protection of any transmission line is primarily consists of fault classification, section identification, and fault location. Fault classification is identifying the type of fault, section identification is finding section of transmission line where the fault has occurred, and fault location is predicting the precise location of the fault. Although all these tasks are intrinsically

difficult, this difficulty is further increased by introduction of series compensation. Different methods have been proposed in literature to overcome problems associated with these steps.

1.4.1 Fault Classification

Fault classification denotes identifying the type of fault. Knowledge of type of fault may be required for other protection algorithms such as zone protection (distance protection), single pole tripping, and fault location. Zone protection means identifying the faulty zone so that remedial action can be taken as soon as possible. Single pole tripping can help in improving transient stability and reliability of the system [14]. Both zone protection and single pole tripping are teal time applications. Therefore, being a prerequisite of real-time application, fault classification algorithm should also run in real-time. However, fault classification can also be used for non-real-time applications like fault location. Moreover, fault classification helps fault-repair crew to estimate men and material required for repairing the faulted line.

In [15], different Kalman filters with different hypothesis were modeled to estimate type of fault. Kalman filter design requires modeling various parts of power system. This is a hard task, particularly modeling the fault resistance. A travelling wave based method [16] uses high-frequency voltage signals. However, capacitive voltage transformers (CVT) can filter out high-frequency component of signal, due to their capacitive nature [17].

Magnitude and angle of fundamental phasors were used in a fuzzy logic based protection technique [18] in an uncompensated line. However, estimation of fundamental phasors during fault in is not accurate. Inaccuracy of phasor estimation is more in the case of series compensated line. Fuzzy logic based methods for compensated line were proposed in [19, 20] where features were extracted from current signals using higher order statistics [19] and discrete wavelet transform [20]. An algorithm based on traveling waves for the protection of series compensated line is proposed in [21] .

Recently a method using combined S-Transform and Logistic Model Tree (ST-LMT) has been proposed in [22]. This method uses S-Transform for pre-processing of the current signal before giving it as input to Logistic Model Tree for fault classification and this has increased encouraging results.

As mapping from voltage and current signals to type of fault is highly nonlinear, last two decades have seen rise in popularity of machine learning based techniques [23-30] due to their capability to map highly nonlinear input output relation. Various artificial neural networks (ANN) based techniques are used with different level of success. A self-organizing map (SOM) is formed in [24] by extracting features from wavelet-transform. For protection of series compensated line radial-basis-function neural network (RBFNN) based technique [23] has shown better results than error back propagation neural network based technique.

Similarly, in [26] trained probabilistic neural network (PNN) used features extracted from S-transform for fault classification. Although training of PNN and RBFNN is very fast, their memory and calculation requirements may prove limiting factor for their implementation in real time application of transmission line protection. Despite good performance of ANN based techniques they require large number of training instances.

Recently, another machine learning approach that is Support Victor Machine (SVM) is getting popularity over ANN due to its advantages over ANN. SVM is a maximum margin classifier because unlike ANN, SVM solves a convex optimization problem. One of the drawback of ANN is that it could stuck in local optima [31]. Unlike ANN, very few parameters are required to adjust for SVM [32, 33]. Firing angle of TCSC and current samples from a-phase were given to SVM for classifying faults of all phases [28]. However, firing angle input to SVM is taken from TCSC, which is 150 km away from relaying end. Improvements over this method were achieved in [29] by giving current samples from each phase to corresponding phase classifier. Similarly, [30] has given features extracted from current samples with wavelet transform to SVM. However, methods of [29, 30] may suffer from overfitting [32, 33] on testing set as no separate validation or cross-validation was performed to find the values of parameters.

1.4.2 Section Identification

Throughout this thesis, the term section identification refers to “identifying (when seen from relay) whether fault has occurred in the section between the relay and the compensating bank or fault has occurred in section between the compensating bank and the opposite end of transmission line.” Section between the relay and the compensating bank is referred as section in front of (or before) compensating bank and section between the compensating bank and other end of transmission line is termed as section behind the compensating bank.

The uncertainty, whether fault is in front of compensating bank or behind the compensating bank, affects the setting of conventional relays. In practical implementations [4] this uncertainty leads to considerable reduction of first zone settings. This can compromise with security of the power system. Therefore, accurate identification of faulty section is critical security of power system.

Faulty section identification can assist zone protection algorithm. For zone protection purpose exact location of fault is not important, only identifying a predefined segment is sufficient. However, to assist a real time application of zone protection, algorithm for faulty section identification must be fast and should have small memory requirements.

Discrete Wavelet Transform (DWT) and Chebyshev Neural Network (ChNN) were used in [34] to detect faulty section. However, ChNN can be slow during section prediction

due to their explicit mapping of input data into higher dimension using Chebyshev expansion. This can leave ChNN unsuitable for real-time zone protection.

With the help of dc component of fault current a fuzzy logic based algorithm [35] was tried to identify faulty section. However, performance for identifying faults in front of compensating bank was very low (85%) due to negligible dc component in some fault cases.

1.4.3 Fault Location

Knowledge of accurate location of fault helps fault-repairing crew to reach the location of fault as soon as possible. This leads to small downtime, which reduces the revenue loss and increases the reliability of power supply. Crew requires fault location information after faulted portion has already been separated by circuit breakers. Therefore, fault location algorithm can run in non-real-time manner and may take few seconds to few minutes. Consequently, memory (storage) and calculation requirements are not as strict as they are with fault classification and section identification. Nevertheless, accuracy of the fault location is the prime concern for fault classification as well as section identification.

Inherently nonlinear problem of fault location further worsens by the nonlinearities introduced by FC, TCSC, and their protective equipment [36]. Phasor values of current and voltage may be unreliable due to transients caused by the presence of series capacitor. Therefore, conventional phasor based methods [37-39] may not be relied upon. Hence, travel time of transients is measured in [40], to locate the fault. In this case, high frequency transients were detected by wavelet transform utilizing synchronized samples from both the ends of transmission line.

1.5 SINGLE-HIDDEN LAYER FEEDFORWARD NETWORKS (SLFNs)

Conventional techniques that were designed for uncompensated lines prove inadequate due to nonlinear nature of the problems. Since artificial neural networks (ANNs) have the capabilities of modeling nonlinearities, their use is widely explored in literature for many power system problems [41] including fault classification, section identification, and fault location normal and TCSC compensated lines [42]. Recently single-hidden layer feedforward networks (SLFNs) are gaining interest due to the advantages offered by them. Specifically four SLFNs are getting special attention due to their ability to map high degree of nonlinearity. These SLFNs are Support Vector Machine (SVM), Relevance Vector Machine (RVM), Extreme Learning Machine (ELM), and Kernel Extreme Learning Machine (KELM).

Support vector machine (SVM) solves convex optimization problem to find model parameters [32, 33]. That is why it usually gives better generalization performance than normal ANN based techniques. In [43] use of SVM has led to reasonable accuracy for fault location. However, training and tuning of parameters (regularization and kernel) for SVM is time-consuming.

Compared with SVM and other ANN based methods, RVM is a relatively new SLFN for classification and regression. RVM [33, 44] uses a sparse Bayesian learning approach, hence it does not have regularization parameter. Moreover, RVMs are highly sparse. Therefore, they are very fast and require small memory, this too without compromising the accuracy. For RVM only parameters related to basis function are required to be adjusted whereas in addition to basis function parameter SVM has a regularization parameter. In the case of ANN many more parameters are required to be adjusted (number of layers and nodes, initial value of weights, learning rate etc.) Although prediction time of RVM is less, RVM has larger training time than SVM. [44-46].

Recently, extreme learning machines (ELMs) [47, 48] are attracting considerable interest due to their smaller training time and comparable or better generalization capability than other SLFNs. Short training time is because the hidden layer does not require tuning. Weights between hidden layer and output layer are found analytically with the help of Moore-Penrose generalized inverse. ELM performed better than SVM for fault location in series compensated line [49]. However, in [49] neither convention of forming ensembles of ELM was followed nor validation (or cross-validation) was performed to avoid overfitting. Overfitting may lead to inflated results.

KELM is an extension of ELM proposed by [50]. KELM have kernel functions as its hidden layer nodes. Similar to ELM, weights on outputs of hidden layer nodes are learned analytically. If tuned properly KELM gives better accuracy than ELM. Tuning the parameters of KELM is much easier than that of SVM and RVM, as its training time is noticeably shorter than SVM and RVM.

This thesis presents new approaches for fault classification, section identification, and fault location in series compensated line using the four popular SLFNs by utilizing nonlinear-mapping capabilities of these SLFNs. Procedures to address these issues are summarized in the following section. Performances of SLFNs are compared with each other based on accuracy, time taken for parameter tuning, training time, and prediction time. The four SLFNs were used for protection of a two-area equivalent of a network and a detailed 12-bus system. Results show that different SLFN has different and competing advantages. Therefore, this thesis explores the suitability different SLFNs for real-time and non-real-time applications.

1.6 MAIN CONTRIBUTIONS OF THE THESIS

The main contributions of the work are summarized as follows:

- To our knowledge, present work is the first attempt to use KELM for fault classification, section identification and location of fault in series compensated line. Moreover, for these purposes, KELM has surpassed performance of other SLFNs in almost every case in terms of accuracy and training time.

- Optimum values of SVM parameters are searched with the help of Genetic Algorithm (GA) for fault classification and identification purpose. Prior to this work, parameters associated with SVM were decided either by trial and error method [28] or by nonlinear-grid search [30, 51] based methods. Use of GA resulted in improvement in accuracy.
- Ten type of shunt faults have been identified by effectively implementing the SVM and RVM classifiers. Their performances are compared on the basis of classification accuracy and testing time. RVM is found to be suitable for online applications as it takes smallest prediction time.
- Faulty section of transmission line is identified by modeling SVM, RVM, ELM, and KELM classifiers. Suitability of each algorithm is checked based on their accuracy and prediction time.
- Regression analysis is performed to predict the accurate location of fault. The four popular SLFNs have been implemented for this purpose. Their performances are measured in terms of their accuracy and training time.
- Another new approach with modified input to SLFN has been proposed for fault location. This method used prior knowledge of faulty section and fault type. This modification of input has further improved accuracy of fault location.
- Two power systems were simulated to validate the efficacy of the proposed methods: 1) Two-area equivalent system, 2) 12-Bus system modeled in detail. These test systems were modeled in PSCAD/EMTDC [52] using inbuilt components and some custom-made components.
- Many fault scenarios were created with wide variation in system and fault parameters e.g. compensation level, load conditions, fault resistance, fault location, and fault inception angle. Current and voltage signals from relaying end were collected for validating the proposed method.
- A new method to automate the process of changing the values of any parameter and invoking the emtp-type program externally is proposed. Here this method is used to create data set for training and testing of SLFNs. However, this method can be used in any transient study where repeated change in parameter is required. It reduces time and efforts involved in many complex power system studies.
- Different types of SLFNs - SVM, RVM, ELM and KELM - are used for fault classification, section identification, and fault location. Their performances are compared on the basis of accuracy, training time, and testing time. It was found that no single algorithm was suitable for all the protection requirements. For real-time applications of fault classification and section identification, SVM or RVM works better and for offline application of fault location KELM performs better.

1.7 ORGANIZATION OF THE THESIS

The overall structure of the thesis takes the form of nine chapters and the work included in each chapter has been presented in the following order:

Chapter 1 introduces the various facets of the topic presented in the thesis and briefly reviews available literature and state of art. This chapter also highlights author's contribution towards the solution of the problem, and finally, outlines the organization of the thesis.

Chapter 2 gives a brief overview of Fixed Capacitor (FC), Thyristor Controlled Series Capacitor (TCSC), and their protective equipment. It also discusses problems posed by introduction of series compensation in transmission line.

Since it is pertinent to know the theory behind the SLFNs before using them to address the problem in hand, **Chapter 3** concisely describes theory behind some popular SLFNs, such as Support Vector Machine (SVM), Relevance Vector Machine (RVM), Extreme Learning Machine (ELM), and Kernel Extreme Learning Machine (KELM).

Chapter 4 describes details of two power systems modeled for verifying the efficacy of the proposed methods. Procedure for generating the data and its distribution for training, validation, and testing are discussed here. Further, a new method for generating large number of voltage and current signals is introduced in this chapter.

Chapter 5 presents methods for fault classification using different SLFNs. Implementation procedure and results are discussed in detail. Parameters of SLFNs are tuned with GA, which yielded better results.

In **Chapter 6**, methods for identification of faulty section are given. Performances of various SLFNs are evaluated on two different power systems.

In **Chapter 7**, accurate location of fault is predicted with the help of various SLFNs. Performances of these SLFNS are compared with each other.

Chapter 8 proposes a new method using KELM for fault location based on faulty-section, type of fault, and modification of inputs to SLFNs according to fault type.

Chapter 9 draws conclusion upon the entire thesis, tying up the various theoretical and practical strands in order to suggest which algorithm is suitable for which part of the problem.

1.8 CONCLUSIONS

Series compensation is a widely accepted option of exploiting full capabilities of existing transmission resources. Series compensation offers many profits. However, benefits bring challenges with them for protection of compensated line. This chapter has briefly elaborated these benefits and challenges. The chapter has given a glimpse of existing research for fault classification, section identification, and fault location. Before offering an overview of contributions of the work and organization of thesis, the chapter has introduced SLFNs that will be explored in the thesis for addressing the problem of fault type identification, faulty section identification, and location of fault.

Before proposing a protection algorithm for series compensated lines, it seems pertinent to know construction and issues related to series compensating devices. Therefore, this chapter begins with a brief overview of Fixed Capacitor (FC), Thyristor Controlled Series Capacitor (TCSC), and their protective equipment. The advantages and applications of series compensation with FC and TCSC are also discussed in this chapter. Moreover, the chapter discusses the challenges posed by introduction of series compensation in transmission lines.

2.1 INTRODUCTION

Fixed series compensation using fixed capacitor (FC) offers many advantages such as better stability margins, reduction of drop in system voltage during severe disturbances, reduction of transmission losses, and improved load sharing on parallel transmission lines. Series compensation becomes further attractive with Thyristor Controlled Series Capacitor (TCSC) by offering a variable level of compensation. However, for same level of compensation TCSC alone becomes uneconomical. Therefore, usually a combination of both FC and TCSC are being used in practical applications [13, 53].

Some classical problems of voltage and current reversal are associated with series capacitors. Moreover, equipment installed to protect FC and TCSC add further complications to it [9, 54]. Typically, FC and TCSC are installed at highly critical lines. Therefore, misoperations of relay on these lines can severely compromise with stability of power system. Consequently, protection of series compensated lines needs extra attention [11, 12]. Thus, before designing protection algorithm for series compensated lines, knowledge of complications faced by series compensation will be valuable. In addition, an overview of FC and TCSC along with their protective equipment seems appropriate at this juncture.

2.2 SERIES COMPENSATION

Series compensation is achieved by adding capacitors in series to a transmission lines to reduce its overall inductive reactance. Addition of series capacitors offers many advantages, including improved stability margins, reduced transmission losses, improved load partition on parallel lines, ability to adjust line load levels and reduced voltage drop in the system during severe disturbances.

The introduction of series compensation is economical for lengths of line greater than 300 km. Nonetheless, it has been effectively applied to lines of shorter length, which is a part of the long transmission system. In general, 25 to 75 percent of the inductive reactance of the line is compensated by using series capacitors.

The reduced series inductance of transmission line achieved by adding capacitors in series provides higher line loading ability and enhanced stability margin. It can be explained easily with the help of the standard power transfer equation for the system shown in Fig. 2.1.

This figure shows two power system connected by series compensated line, where X_{S1} and X_{S2} are equivalent impedance of two systems.

The standard power transfer equation can be expressed as:

$$P = \frac{E_1 E_2 \sin\delta}{X_L - X_C} \qquad (2.1)$$

where δ is the angle between the line ends voltages E_1 and E_2, X_L is the inductive reactance of transmission line and X_C is the reactance of the capacitor and P is the real power transferred. The total transfer impedance is reduced, because of the introduction of capacitive reactance (X_C) introduced by the series capacitor. Therefore, for the same voltages and angle, series capacitor allows more power to transfer through the line. Also, for the same power transfer level a reduction in the system angular separation is required which increases stability margins significantly. For simplicity, the system resistance has been neglected in the above equation, which should be considered during the determination of optimum level of the compensation.

The effect of addition of series capacitors is to increase the fault current levels significantly, which is very harmful to the system. However, the protective equipment of the capacitor are used to bypass the capacitor for such high levels of fault current.

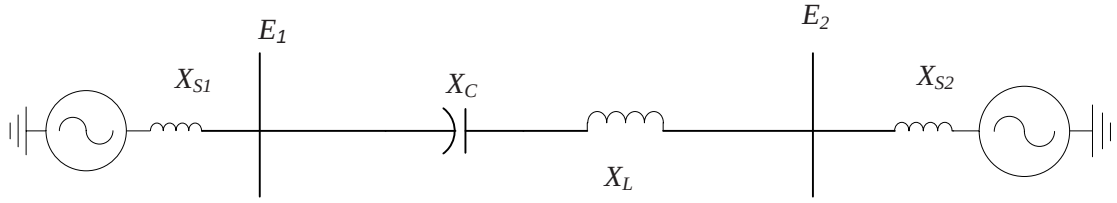


Fig. 2.1: Simple two-area system with series compensation.

2.3 ADVANTAGES OF FC AND TCSC

Fixed Capacitors (FCs) can increase base power flow and loadability of line. However, compensation level cannot be controlled and power transfer may exceed the thermal limit in case of outage of other line in the network. TCSCs can overcome these limitations of FCs by providing fast and continuously controllable compensation, in addition they offer the following advantages:

- Improve voltage stability by compensating part of line reactance, thereby effectively acting as a stiff voltage source for the load.
- For same power flow in line, series compensation can achieve better transient stability margin than uncompensated line.
- TCSC can improve stability of the system by damping the power swings by varying compensation levels in a manner that neutralizes accelerating and decelerating swings of the system.

- Control of power flow in particular lines helps in minimizing losses in the overall system by avoiding loop flow of power and achieving optimal power flow conditions.
- At subsynchronous frequencies, the parallel combination of capacitor and inductor (TCSC) intrinsically tends to be inductive. This inductive nature along with system resistance helps in damping out subsynchronous oscillations. Further damping of oscillations can be accelerated by TCSC control strategies.
- The dc-offset-voltage across capacitor appears during its insertion in line. This dc-offset can be nullified swiftly by controlling firing angle of TCSC thyristors.
- Fast bypass of capacitor is possible with thyristor firing control to protect TCSC from overvoltages. Similarly, rapid reinsertion of a capacitor after the fault can help in maintaining system stability.
- Operation of TCSC can be changed quickly from capacitive to inductive mode to limit high current during short circuit faults.

Although TCSC have so many advantages over FC, they are costlier than FC. Therefore, in practice both FC and TCSC are installed together. Thereby, the advantages of both can be availed simultaneously.

2.4 CHALLENGES POSED BY FC AND TCSC

Despite these advantages, series compensation poses challenges for protection of transmission line. Proper understanding of these problems is desirable before discussing any protection method. Therefore, this section discusses some of these problems.

For the sake of simplicity in subsequent discussion, during fault impedances are assumed to be either purely capacitive or inductive, and thus there is no load flow across the system. However, in actual system load flow can be a significant factor in the performance of the distance protection on or near series compensated lines. This must be taken into account while designing the protection scheme to assure proper coordination.

2.4.1 Over Voltage Protection of Series Capacitor (Protective Equipment)

The series capacitors face a wide range of currents, which can result in large voltages across the capacitors. Above it, during short circuit faults, high current may lead to dangerously high voltage. In general, it is uneconomical to design the capacitors that can tolerate these overvoltages. Therefore, to protect the capacitor from over voltage protective equipment are installed, such as Metal Oxide Varistor (MOV), triggered air gap, and bypass circuit breakers. During high fault currents, one of this equipment may bypass the series capacitor partially or fully. Settings of the relay should be adjusted according to whether the capacitor is conducting fault current or bypassed. Evidently, the characteristics of MOV are highly critical for protection of series compensated line.

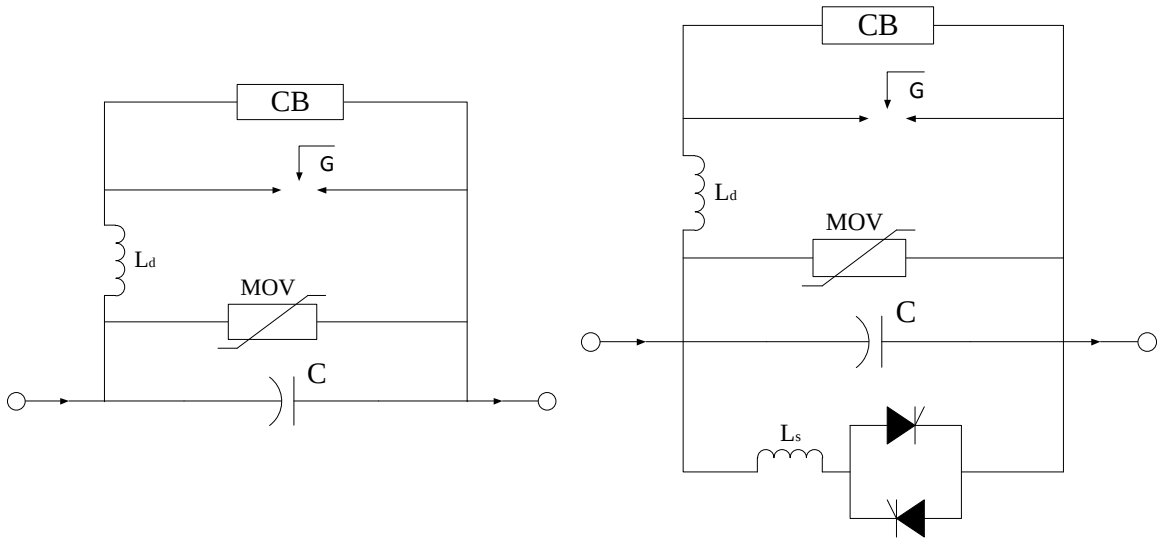


Fig. 2.2: Construction of (a) Fixed Capacitor (FC) and (b) TCSC**.**

Initially parallel power gaps were used to protect the capacitors from overvoltages, and then during 1980s MOVs have replaced parallel power gaps. Fig. 2.2 (a) shows a simplified schematic diagram of this kind of arrangement for Fixed Capacitor (FC), where the MOV is connected directly across the capacitor (C) with no intervening reactor or gap. Moreover, one triggered air gap (G) and a circuit breaker (CB) is connected to bypass the capacitor and MOV, if current raises beyond protection level of MOV. A damping reactor (L_d) is connected to prevent high and abrupt discharge of capacitor (C). In addition of these arrangements, TCSC (shown in Fig. 2.2 (b)) has a controllable reactor (L_s) in parallel with the capacitor (C). Impedance of reactor is controlled by controlling firing pulses of antiparallel pair of thyristors connected in series with reactor.

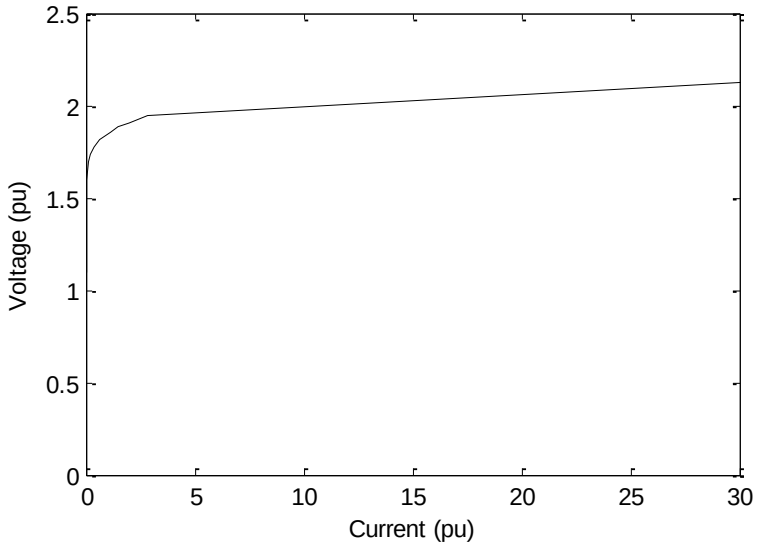


Fig. 2.3: Operating characteristic of MOV.

MOV have nonlinear characteristic as shown in Fig. 2.3. This characteristic insures no current conducts during normal mode of operation, while start conduction rapidly during faulted condition. A fault in the system causes rise in voltage across the capacitor and when this voltage level reaches a pre-defined level, MOV starts conduction to clamp the voltage to the protective level. In each half cycle this clamping action occurs with conduction interchanging between the capacitor and the MOV. After fault cleared, the MOV returns back to its normal conduction-free mode of operation, except for the occasional peak voltage of the post fault transient.

The MOV has a stable and defined characteristic of voltage and energy. A parallel gap is provided to protect the MOV when its energy rating reaches the limit. The energy duty of the MOV is monitored by protection circuit that triggers the parallel gap.

There are many ways to specify the energy rating of the varistor for internal line faults. The MOV is monitored by the varistor protection circuits, which trigger the gap of detection of high fault level. Usually triggering of gap takes few cycles. However, in some cases MOV has the ability to withstand severe three-phase fault for 5 cycles. Thus, MOV is hardly bypassed for an internal fault. For any of these conditions, it is highly improbable that gap will be triggered in the first half cycle of fault. In this work, postfault half cycle current samples are used for fault classification, section identification and fault location. Because gap and bypass switch will have no effect in the first half cycle, they are not considered in simulation studies.

Depending upon the type and magnitude of the fault, MOVs can conduct asymmetrically i.e. MOV of one of the phases may conduct and bypass capacitor partially or fully, while capacitor of another phase may remain in the circuit. This adds further complexity to the protection of the line.

2.4.2 Transients during Faults

Conventionally relays take decision by using functions that use fundamental component of the signals. However, phasor estimation during fault becomes difficult on compensated lines due to transients introduced by series capacitor.

In uncompensated lines, a decaying dc is the main transient during a fault. However, in series compensated lines the main transient is an ac signal. The frequency of this ac signal can be given by:

$$f_e = \frac{1}{2\pi\sqrt{LC}} = f\frac{X_C}{X_L} \tag{2.2}$$

where f is system frequency, X_C is the reactance of the capacitor, and X_L is inductive reactance of the rest of the system including line reactance. If a fault is after capacitor and near to capacitor then X_C may be greater than X_L, in this case transients with frequency

higher than the system frequency may distort the signal. However, higher frequency components may not be generated, if high currents during faults bypass the series capacitor by operating capacitors' protective equipment. Whereas, for faults at remote end X_C may be lesser than X_L then lower frequency (subsynchronous) transients corrupt measurement of fundamental phasor. Moreover, at lower frequency capacitive reactance increases while inductive reactance decreases in TCSC. Therefore, compensation appears to be more than desired compensation level. Similarly, for higher frequency transients, compensation appears to be less than the desired level.

2.4.3 Subharmonics

The presence of the capacitor and inductance of the system forms a series resonant circuit. The frequency of this resonant circuit (neglecting resistance) can be obtained by

$$f_e = \frac{1}{2\pi\sqrt{LC}} = f\,\frac{X_C}{X_L} \qquad (2.3)$$

where f is system frequency, X_C is the reactance of the capacitor, and X_L is inductive reactance of the rest of the system including line reactance. Since, the typical value of X_C/X_L lies in the range of 0.25 to 0.75, f_e will be a less than f i.e. it will be subharmonic of the power frequency.

Many system disturbances cause the excitation of the system at the subharmonic frequency, which in turn may result in transient currents. These transients are usually damped out after a few cycles. Nevertheless, occasionally subharmonic transients may last significantly longer.

Sometimes, the interaction of the subharmonic transient currents with the generators in the system may create problem. The transients in the generator(s) may reflect as a negative slip frequency on the rotor, consequently this appears as net negative impedance into the system. If the total positive impedance of the system becomes lower than this negative impedance, then if corrective action is not taken the transient currents may raise beyond acceptable limits. However, series capacitor will be protected by protective equipment (MOV etc.) if current increase beyond the protective level.

2.4.3.1 Subsynchronous resonance

The natural torsional frequencies of the mechanical shaft system of the generator(s) may be excited by the transients in the system. This complicated phenomenon is termed as subsynchronous resonance (SSR). Depending upon the degree of damping and resonance, the torsional oscillations may be severe enough to cause damage to the shaft. Generators with a large steam turbine that have 2 to 5 torsional modes are most vulnerable to SSR, their torsional modes are normally in the range of 5 to 55 Hz (for system frequency of 60 Hz).

Usually, before installation of series capacitor subharmonic transients and subsynchronous resonance are analyzed and corrective actions are taken to alleviate their effects [8].

2.4.4 Change in Impedance seen by Relay

Normally impedance seen by relay increases with an increase in distance of the fault from the relay. However, when the capacitor is installed in-between there is a sudden decrease in impedance seen at the point of capacitor installation. Therefore, conventional impedance relays overreach or faults just after capacitor can be seen as faults in reverse direction. Fig. 2.4 shows line reactance seen by relay along with fault location on the transmission line. Impedance seen from point A (relaying bus) increases with increase in distance of fault location until point B, just after this point reactance seen becomes negative due to installation of capacitor at this point. Therefore, fault can be seen as in the opposite direction until point C, after which reactance again becomes positive.

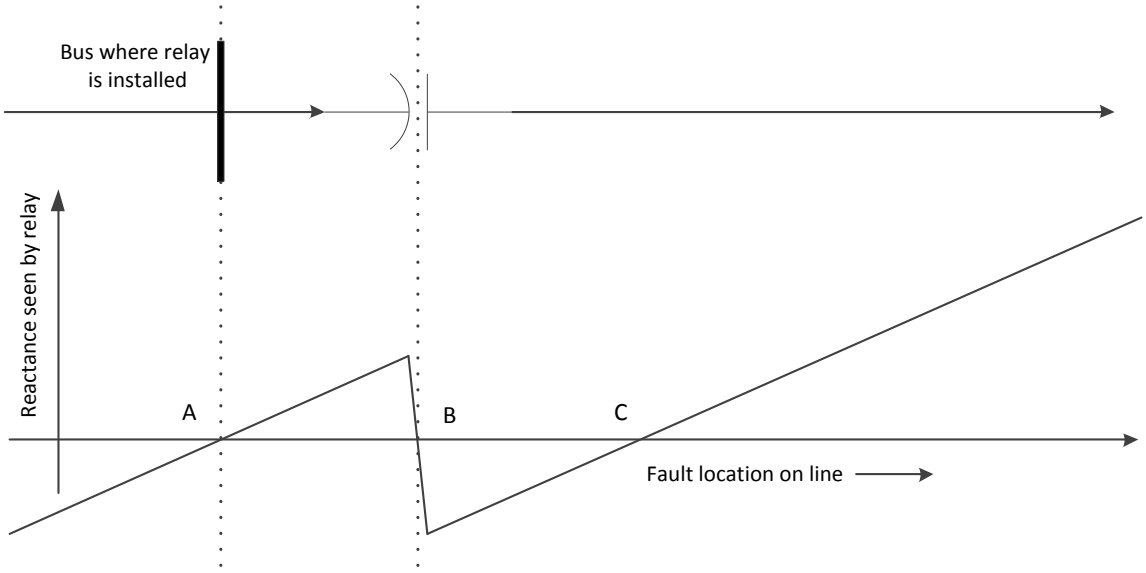


Fig. 2.4: Change in impedance seen by relay at the point of series capacitor installation.

2.4.5 Voltage Reversals

If reactance seen up to faults is capacitive, voltage reversal will take place i.e. the voltage seen by relay will be 180 degree out of phase to voltage of uncompensated line. Normally this happens when a fault is near to capacitor installed. As seen from relaying point A in Fig. 2.4, faults occurring between B and C will face the condition of voltage reversal. Generally, distance algorithms are intended for inductive system. Therefore, protection algorithms will require a revision in this case.

2.4.6 Current Reversals

Current inversion is a phenomenon observed in series compensated lines, if the location of fault in a transmission line is such that system on one side of the fault is

inductive and on opposite side system is capacitive. In such a situation current seems to be flowing from one end of the line and exiting from the other end. In such cases a fault internal to protected transmission line may look like an external fault.

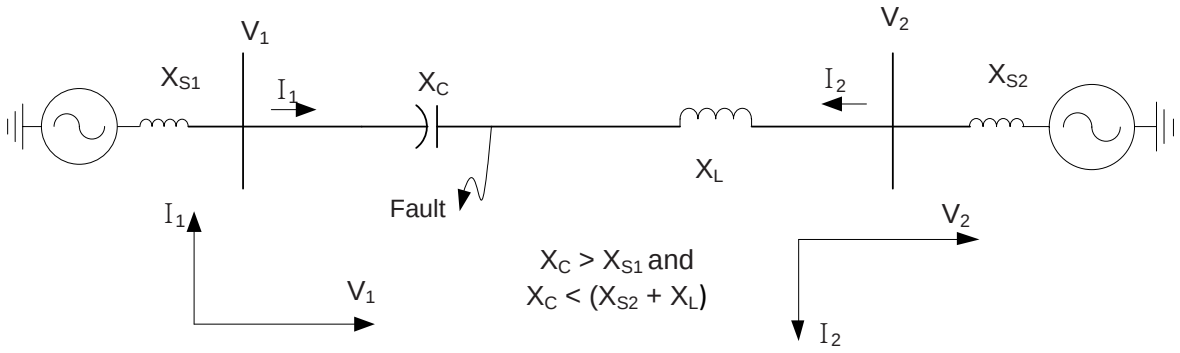


Fig. 2.5: Current inversion on transmission line.

As shown in Fig. 2.5, if the inductive reactance of source (X_{S1}) is less than the capacitive reactance X_C and on the right hand side of fault reactance (X_{S2}+ X_L) is inductive. Therefore, on the left side of the fault, current I_1 leads voltage V_1 and on the right side, I_2 lags voltage V_2. Consequently, I_1 and I_2 are nearly opposite in phase with each other. It may not be possible for very high current fault as then MOV will bypass the capacitor. However, if fault current is low due to high fault resistance, then MOV may not conduct and current inversion can happen.

If system reactance (X_{S1}) on the left side of capacitor is nearly equal to the reactance of the capacitor (X_C), then current on the right become nearly zero. This creates a problem for detection of type of fault and other protection problems. Negative and zero sequence networks can also face current inversion.

Current inversion can affect phase comparison, distance, directional, and differential elements corresponding to phase or sequence-component.

2.4.7 Operating Modes of TCSC

The TCSC's control circuit should be such that it must take required protective action during the fault in the system. Normally, the firing angles of the TCSC's control circuit take it into inductive mode to decrease fault current level. For large short circuit current the MOV comes into reduced resistance mode and then produces an equivalent impedance of TCSC in bypassed inductive mode. Thus, capacitive reactance of TCSC changes gradually into inductive reactance. This change in reactance undesirably affects protection scheme settings of the transmission line [9, 55]. For small fault current, the thyristor is not bypassed by its control circuit, which makes impedance characteristics more complex. In addition, during unsymmetrical waveforms of transient voltage and current, firing circuit of TCSC

does not behave properly. In this case, harmonics generated by TCSC becomes very complex [10].

An understanding of normal operating modes of TCSC is required before understanding the operation of TCSC during faults. Typically, TCSC operates in one of the following modes [9, 54].

- Capacitive-vernier mode ($\alpha_{Lim} < \alpha < \pi/2$): TCSC behaves as a variable capacitor.
- Inductive-vernier mode ($0 < \alpha < \alpha_{Lim}$): TCSC behaves as a variable reactor.
- Bypassed-Thyristor mode ($\alpha = 0$): Thyristors conduct fully and overall reactance is inductive.
- Blocked-Thyristor mode ($\alpha = \pi/2$): Thyristors stop conducting fully and only capacitor remains in the circuit.

Depending upon the type of fault, TCSC gives different operation during the fault. The various conditions of TCSC during fault are described here in brief.

2.4.7.1 Capacitive-vernier mode ($\alpha_{Lim} < \alpha < \pi/2$) without MOV conduction

A fault with high impedance causes a low fault current to flow in the system. This low fault current results in lower voltage level across the TCSC than its preset protective voltage level. Under this condition, the fault current passes through TCSC while the MOV continue to be in its high impedance mode. Due to the presence of compensator in the transmission line, the protective relays can overreach significantly, additionally its directional integrity can be lost.

2.4.7.2 Inductive-vernier mode ($0 < \alpha < \alpha_{Lim}$) with MOV conduction

For a high fault current, the MOV reacts within a half-cycle to decrease the voltage across TCSC. In this case, the capacitor remains un-shorted by both the MOV and circuit breaker. During the fault, the short circuit condition of this type is usually repeated several times. During this condition, the TCSC impedance forms a parallel combination of the capacitor with the low resistance MOV. The protective relay may overreach differently in this case as in the previous case without MOV conduction.

2.4.7.3 Bypassed-thyristor mode ($\alpha = 0$)

A very high short circuit current leads towards total thyristor conduction as the MOV operation will not be enough to decrease the voltage across the capacitor. In this case, the presence of reactor in the circuit may cause the protecting distance relay to underreach.

2.4.7.4 Blocked-thyristor mode ($\alpha = \pi/2$)

During the transient caused by the fault, rapid change in the phase angle of the voltage across the capacitor causes the firing angle of TCSC to change quickly in some

cases. This kind of situation is avoided by blocking the firing mechanism of the thyristor, during this time TCSC behaves as a fixed capacitor. In this case, overreach may occur similar to the case of fixed capacitor with MOV. However, this overreach is less severe than overreach of the TCSC is in the capacitive boost mode.

2.4.7.5 Circuit breaker bypass mode

In case the fault is not cleared within a preset time, the TCSC transfers to breaker bypass mode. Because the series reactor in the circuit breaker is small, the protective relay experiences the normal situation. This condition is used only in providing backup protection.

2.5 CONCLUSIONS

Advantages of series compensation come with many challenges. This chapter has given the details of these advantages and challenges. Conventionally, protection of series compensated lines with fixed capacitor is a complicated task. This becomes further complicated with TCSC due to its variance in compensation level and different operating modes of TCSC during faults. In general, issues that create difficulty for fault classification, section identification, and fault location are discussed in this chapter.

CHAPTER 3: SINGLE-HIDDEN LAYER FEEDFORWARD NETWORK (SLFN)

This chapter concisely describes the theory behind four popular single-hidden layer feedforward network (SLFNs), namely Support Vector Machine (SVM), Relevance Vector Machine (RVM), Extreme Learning Machine (ELM), and Kernel Extreme Learning Machine (KELM). Then chapter goes on to describe methods used to tune parameters related to models of these SLFNs. At the end, it compares all these methods in a general sense.

3.1 INTRODUCTION

Problems of fault classification, section identification, and location in series compensated lines are highly nonlinear in nature. Artificial Neural Networks (ANNs) are known to have excellent capabilities of mapping nonlinearities from input to output. Therefore, ANNs have been used successfully where such nonlinearities exist in power system applications [56-60]. There are numerous kind of ANNs, among them feedforward networks are probably most common type of ANNs. Feedforward networks have one input layer, one output layer, and between input and output layers there are one or multiple layers known as hidden layers, because they are hidden from inputs and outputs. Past two decades have seen a steep rise in the use of single-hidden layer feedforward networks (SLFNs), mainly Support Vector Machine (SVM), Relevance Vector Machine (RVM), Extreme Learning Machine (ELM), and Kernel Extreme Learning Machine (KELM).

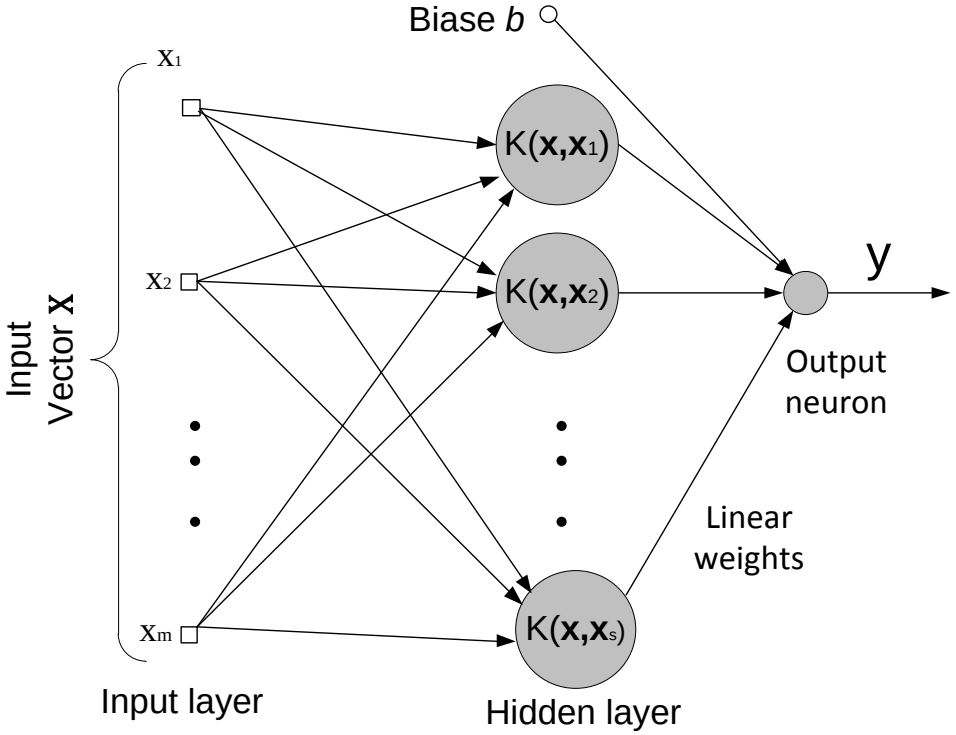


Fig. 3.1: Single-hidden layer neural network (SLFN).

Fig. 3.1 show a model of SLFN, input is given to the input layer and output is receive at output layer, there is single layer that is hidden from input as well as output. Generally,

hidden layer of SLFN has nodes with nonlinear activation function, to enable them to map nonlinearities, as single linear hidden layer cannot map nonlinearities [32, 33, 61]. In Fig. 3.1, these nonlinear nodes are represented by $K(\mathbf{x}_1,\mathbf{x}_2)$ which is referred as kernel-function.

Support Vector Machine (SVM) is possibly one of the most used SLFN in last one and half decade [32, 33, 62]. Based on the principle of structural risk minimization, SVM has better generalization capacity and lesser parameters tuning requirement than conventional ANNs. Power system community has actively utilized SVM for different applications [63-67]. However, higher number of support vectors (SVs) makes it slower during testing. Relevance Vector Machine (RVM) overcame these limitations of SVM by creating a sparse model [33, 44, 45]. Similar to SVM, RVM is also marking its utility in various power system applications [68]. However, RVM has even larger training time than SVM. Parameter tuning of RVM and SVM requires extra efforts.

Extreme Learning Machine (ELM) is a relatively new SLFN, which does not require parameter tuning and has very small training time [47, 48, 69]. However, because hidden layer nodes of ELM are assigned randomly, ELM may lead to overfitting and therefore ELM may require technique such as making ensembles. Kernel Extreme Learning Machine (KELM) is a recent improvement over ELM, by restoring to the kernel function for its formulation [50].

Work in this thesis uses these SLFNs namely SVM, RVM, ELM, and KELM. Therefore, this chapter gives a brief theory of these four SLFNs to understand their efficient utilization and to find their suitability for a particular application.

3.2 SUPPORT VECTOR MACHINE (SVM)

Here SVM is introduced briefly, detailed theory behind SVM can be found in [32, 33, 62, 70]. Formulation of SVM for classification and regression are slightly different. Therefore, they need separate treatment. Work presented in this thesis used libsvm-mat-2.91-1 [71] toolbox in MATLAB to cross-validate, train and test SVM-models.

3.2.1 SVM Classification

Assuming a training set $\{\mathbf{x}_i, y_i\}_{i=1}^{N}$, where $\mathbf{x}$ is input vector and y is corresponding output class label, an N is total number of training instances. Normally a discriminant function is designed to formulate a classification problem. The discriminant function of SVM is chosen as:

$$f(\mathbf{x}) = \mathbf{w} \cdot \boldsymbol{\varphi}(\mathbf{x}) + b$$

where $\mathbf{w}$ is weight vector, b is bias, and $\boldsymbol{\varphi}(\mathbf{x})$ is a mapping function to map the input vector $\mathbf{x}$ into higher dimensional space H. This discriminant function represents a hyperplane in H.

SVM tries to find an optimum hyperplane that is at maximum distance from points (vectors) of both classes. Hence, it is known as a maximum margin classifier. Points that decide the position of optimal hyperplane and are nearest to it (located on supporting hyperplane) are called support vectors. This is shown pictorially in Fig. 3.2 for tow dimensional input data. Here data points belonging to two classes are represented by circles and crosses respectively.

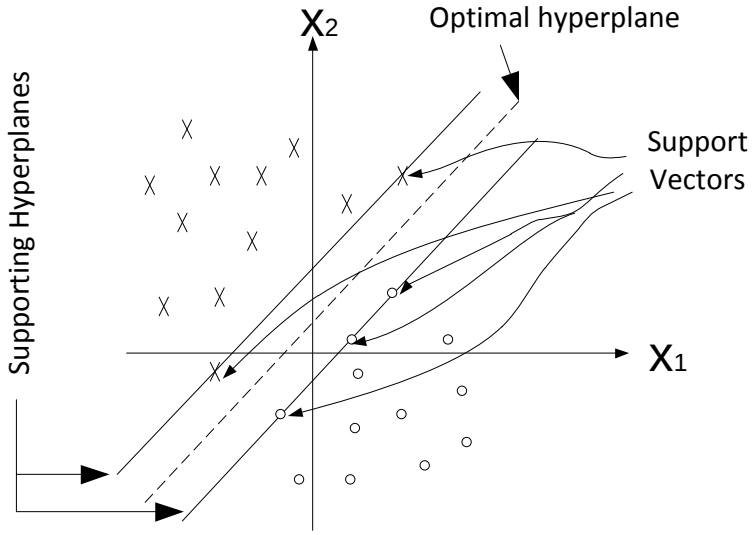


Fig. 3.2: SVM as maximum margin classifier.

Following optimization problem can be formulated to find the value of **w** and b:
Minimize:

$$J(\mathbf{w},\xi)=\frac{1}{2}\|\mathbf{w}\|^2+C\sum_i \xi_i \tag{3.1}$$

Subject to:

$$y_i(\mathbf{w}\cdot\boldsymbol{\varphi}(\mathbf{x}_i)+b)\geq 1-\xi_i \qquad for\ \xi_i\geq 0, and\ i=1,2,\ldots,l \tag{3.2}$$

where C is regularization parameter and ξ_i are measure of error in case of nonseparable data points and y_i is the class label of i^{th} data point.

Dual of this optimization problem is given by:
Maximize:

$$L_D=\sum_i \alpha_i-\frac{1}{2}\sum_{i,j}\alpha_i\alpha_j y_i y_j K(\mathbf{x}_i,\mathbf{x}_j) \tag{3.3}$$

Subject to:

$$0\leq \alpha_i \leq C \tag{3.4}$$

$$\sum_i \alpha_i y_i=0 \tag{3.5}$$

where, α_i's are Lagrange multipliers, $K(\mathbf{x},\mathbf{y})$ is known as kernel function, it is a non-linear function and defined as

$$K(\mathbf{x},\mathbf{y}) = \boldsymbol{\varphi}(\mathbf{x}).\boldsymbol{\varphi}(\mathbf{y}) \tag{3.6}$$

Therefore, after obtaining optimal values of Lagrange multipliers (α_i^*), discriminant function can be written as

$$f(\mathbf{x}) = \sum_{i=1}^{N_S} y_i \alpha_i^* K(\mathbf{x}_i, \mathbf{x}) + b^* \tag{3.7}$$

where, N_S is the number of support vectors, these support vectors correspond to nonzero values of Lagrange multipliers (α_i^*), other vector have zero value of Lagrange multipliers for them and therefore have no role in deciding optimal hyperplane (Fig. 3.2).

$$\begin{array}{lll} If & f(\mathbf{x})\big|_{\alpha^*} < 0 & class-I \\ If & f(\mathbf{x})\big|_{\alpha^*} > 0 & class-II \end{array}$$

3.2.2 SVM Regression

Consider a set of training data $\{\mathbf{x}_i, t_i\}_{i=1}^N$ and a generic regression function for support vectors machine [32, 33, 72] can be defined as:

$$f(\mathbf{x}) = \mathbf{w} \cdot \mathbf{x} + b \tag{3.8}$$

where, weight vector $\mathbf{w} \in R^n$, and bias $b \in R$. Our goal is to find the value of the weight vector ($\mathbf{w}$) and bias (b) such that the values of $\mathbf{x}$ can be determined by minimizing the regression risk:

$$R_{reg}(f) = C\sum_{i=0}^{N} \xi(f(\mathbf{x}_i) - t_i) + \frac{1}{2}\|\mathbf{w}\|^2 \tag{3.9}$$

Subject to:

$$\begin{array}{ll} (\mathbf{w} \cdot \mathbf{x}_i + b) - t_i \geq \varepsilon + \xi_i & for\ \xi_i \geq 0, and\ i = 1, 2, \ldots, N \\ t_i - (\mathbf{w} \cdot \mathbf{x}_i + b) \geq \varepsilon + \hat{\xi}_i & for\ \hat{\xi}_i \geq 0, and\ i = 1, 2, \ldots, N \end{array} \tag{3.10}$$

where C is Regularization Parameter and $\xi(.)$ is a loss function, and ϵ is parameter of ϵ-insensitive loss function i.e. the error between target point and predicted point is neglected if it is less than ϵ **(**shown in Fig. 3.3**)**.

Dual formulation of (3.9) leads us to maximization of

$$-\frac{1}{2}\sum_{i,j=1}^{N}(\alpha_i - \alpha_i^*)(\alpha_j - \alpha_{jj}^*)(\mathbf{x}_i \cdot \mathbf{x}_j) - \varepsilon\sum_{i=1}^{N}(\alpha_i + \alpha_i^*) + \sum_{i=1}^{N} y_i(\alpha_i - \alpha_i^*) \tag{3.11}$$

Subject to:

$$\sum_{i=1}^{N} \alpha_i - \alpha_i^* = 0, \quad \alpha_i, \alpha_i^* \in [0, C]$$

The Lagrange multipliers, α_i and $\alpha^*_{i,}$ represent solutions to the above quadratic problem. Only the non-zero values of the Lagrange multipliers in equation (3.11) are useful in forecasting the regression line and are known as support vectors.

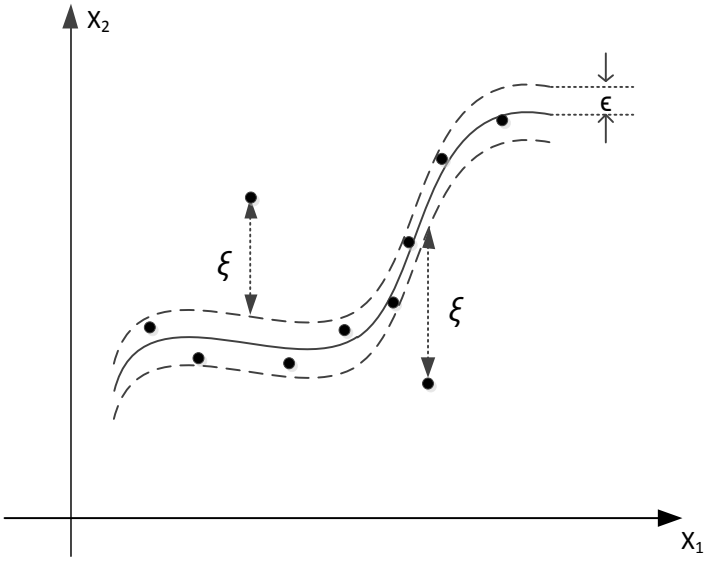


Fig. 3.3: Support vector regression with ϵ-insensitive loss function.

After solving the dual optimization problem of (3.11), we get the weight vector

$$\mathbf{w} = \sum_{i,j=1}^{N} (\alpha_i - \alpha_i^*)\mathbf{x}_i$$

Therefore, the function given by (3.8) becomes

$$f(\mathbf{x}) = \sum_{i,j=1}^{N} (\alpha_i - \alpha_i^*)(\mathbf{x}_i \cdot \mathbf{x}) + b \qquad (3.12)$$

To convert this linear regression into nonlinear regression the dot product $\mathbf{x}_i \cdot \mathbf{x}$ in (3.12) can be replaced with the kernel function $K(\mathbf{x}_i, \mathbf{x})$ given in (3.30)

$$f(\mathbf{x}) = \sum_{i,j=1}^{N} (\alpha_i - \alpha_i^*)K(\mathbf{x}_i, \mathbf{x}) + b$$

3.3 RELEVANCE VECTOR MACHINE (RVM)

RVM maps the functionalities of SVM, i.e. RVM considers a function *y* (similar to SVM) of the following form:

$$y(\mathbf{x}) = \sum_{i=1}^{N} w_i K(\mathbf{x}_i, \mathbf{x}) + w_0 \qquad (3.13)$$

However, their training procedure of RVM is different form SVM. RVM [33, 44] uses Sparse Bayesian Prior distribution over weights to estimate the parameters from the training data. MATLAB Toolbox of Sparsebayes (Version 2.00) [73] has been used in this work to train and test RVM models. Since the formulation of RVM classification builds upon the formulation of RVM regression, initially theory of regression will be discussed.

3.3.1 RVM Regression

With a given training data set $\{\mathbf{x}_i, t_i\}_{i=1}^{N}$, assuming targets ($t$) are sampled with an additive Gaussian distributed noise (ε) that has zero mean and standard deviation σ.

$$t_i = y(\mathbf{x}_i, \mathbf{w}) + \varepsilon_i \tag{3.14}$$

$$p(\varepsilon_i) = \mathcal{N}(\varepsilon \mid 0, \sigma^2) \tag{3.15}$$

Hence,

$$p(t_i \mid \mathbf{x}_i, \mathbf{w}) = \mathcal{N}(t_i \mid y(\mathbf{x}_i, \mathbf{w}), \sigma^2) \tag{3.16}$$

where, $\mathbf{x}$ is training data point, $\mathbf{w}$ is weight vector, and $\mathcal{N}$ denotes Normal/Gaussian Distribution.

Sparse Bayesian Prior distribution over weights gives

$$p(\mathbf{w} \mid \boldsymbol{\alpha}) = \prod_{i=1}^{M} \mathcal{N}(w_i \mid 0, \alpha_i^{-1}) \tag{3.17}$$

where $\boldsymbol{\alpha} = [\alpha_1\ \alpha_2\ \ldots\ \alpha_M]^T$ is a vector of M hyperparameters, that are assumed as independent random variables. A *Gamma* prior distribution is considered on these hyperparameters

$$p(\alpha_i) = Gamma(a, b) \tag{3.18}$$

Now σ and $\boldsymbol{\alpha}$ are learnt from training data using Bayesian learning technique. Once $\boldsymbol{\alpha}$ is learnt weights are also known. For most of the weights, posterior distribution becomes sharply peaked around zero i.e. weights become zero except relevant weights corresponding to relevant vectors. Hence, the function given in (3.13) becomes:

$$y(\mathbf{x}) = \sum_{i=1}^{N_R} w_i^* K(\mathbf{x}_i, \mathbf{x}) + w_0^* \tag{3.19}$$

where, N_R is number of relevance vectors and w_i^* are inferred values of the weights. Now outputs (t) can be predicted from (3.16) using calculated weights.

3.3.2 RVM Classification

For two-class (binary) classification, function given in (3.13) is generalized by applying the logistic sigmoid link function as follows:

$$\rho(y) = \frac{1}{1 + e^{-y}} \tag{3.20}$$

For classification, instead of the Gaussian distribution, Bernoulli distribution is adopted for $P(t|\mathbf{x})$. Therefore, the likelihood can be written as:

$$P(t \mid \mathbf{w}) = \prod_{n=1}^{N} \rho\{y(\mathbf{x}_n; \mathbf{w})\}^{t_n} [1 - \rho\{y(\mathbf{x}_n; \mathbf{w})\}]^{1-t_n} \tag{3.21}$$

where, targets are binary $t_n \in \{0,1\}$. Unlike regression in RVM-classification no close form solution can be found therefore iterative methods [45, 74] are adopted to find the weights.

3.4 Extreme Learning Machine (ELM)

Extreme learning machines are single hidden layer feedforward networks (SLFNs) with random hidden nodes [48]. Consider a dataset with *N* samples $\{\mathbf{x}_i, \mathbf{t}_i\}_{i=1}^{N}$ and a typical SLFN with *L* hidden nodes and activation function. $g_i(\mathbf{x}_j)$. Here $\mathbf{x}_i = [x_{i1}, x_{i2}, ..., x_{in}]^T$ and $\mathbf{t}_i = [t_{i1}, t_{i2}, ..., t_{im}]^T$. The output nodes are linear and the output $\mathbf{o}_j$ can be expressed as

$$f_j(\mathbf{x}) = \sum_{i=1}^{L} \beta_{ij} h_i(\mathbf{x}) = \mathbf{h}(\mathbf{x})\beta_j \qquad \text{for } j = 1, \ldots N. \tag{3.22}$$

where $\beta_j = [\beta_{1j}, \beta_{1j}, ...\beta_{Lj}]^T$ is the weight vector between hidden node and j^{th} output node and $\mathbf{h}(\mathbf{x}) = [h_1(\mathbf{x}), h_2(\mathbf{x}), \ldots, h_L(\mathbf{x})]$ is output (row) vector of the hidden layer for input $\mathbf{x}$.

For the better generalization performance of feedforward neural network, its error on the training set and norms of the weights should be minimum [75]. Therefore, the training error along with the norm of the output weights is minimized in ELM i.e.

Minimize: $\|\mathbf{H}\boldsymbol{\beta} - \mathbf{T}\|^2$ and $\|\beta\|$ (3.23)

where **H** is the hidden-layer output matrix

$$\mathbf{H} = \begin{bmatrix} \mathbf{h}(\mathbf{x}_1) \\ \vdots \\ \mathbf{h}(\mathbf{x}_N) \end{bmatrix} = \begin{bmatrix} h(\mathbf{x}_1) & \cdots & h(\mathbf{x}_1) \\ \vdots & \ddots & \vdots \\ h(\mathbf{x}_N) & \cdots & h(\mathbf{x}_N) \end{bmatrix}_{N \times L}, \quad \boldsymbol{\beta} = \begin{bmatrix} \beta_1^T \\ \vdots \\ \beta_L^T \end{bmatrix}, \text{ and } \quad \mathbf{T} = \begin{bmatrix} \mathbf{t}_1^T \\ \vdots \\ \mathbf{t}_N^T \end{bmatrix}$$

Given a training set, the number of hidden nodes and hidden node activation functions, the originally proposed ELM [48] given solution as,

$$\beta = \mathbf{H}^{\dagger}\mathbf{T} \tag{3.24}$$

where $\mathbf{H}^{\dagger}$ is the Moore–Penrose generalized inverse of the matrix **H**.

3.5 Kernel-ELM

Huang et el. [50] further generalized solution of ELM given by (3.24) as:

$$\beta = \mathbf{H}^T \left(\frac{\mathbf{I}}{C} + \mathbf{H}\mathbf{H}^T \right)^{-1} \mathbf{T}$$

$$\mathbf{f}(\mathbf{x}) = \mathbf{h}(\mathbf{x})\beta = \mathbf{h}(\mathbf{x})\mathbf{H}^T \left(\frac{\mathbf{I}}{C} + \mathbf{H}\mathbf{H}^T \right)^{-1} \mathbf{T} \tag{3.25}$$

where *C* is a regularization parameter. This gives the smallest norm least-squares solution of the above linear system and this solution is unique.

Now Kernel matrix can be defined using Mercer's conditions for unknown feature mapping h(x).

$$\boldsymbol{\Omega}_{\text{ELM}} = \mathbf{H}\mathbf{H}^T : \Omega_{\text{ELM}\,i,j} = h(\mathbf{x}_i) \cdot h(\mathbf{x}_j) = K(\mathbf{x}_i, \mathbf{x}_j)$$

Therefore (3.25) can be written as

$$\mathbf{f}(\mathbf{x}) = \mathbf{h}(\mathbf{x})\mathbf{H}^T \left(\frac{\mathbf{I}}{C} + \mathbf{H}\mathbf{H}^T \right)^{-1} \mathbf{T}$$
$$= \begin{bmatrix} K(\mathbf{x}, \mathbf{x}_1) \\ \vdots \\ K(\mathbf{x}, \mathbf{x}_N) \end{bmatrix}^T \left(\frac{\mathbf{I}}{C} + \mathbf{\Omega}_{\text{ELM}} \right)^{-1} \mathbf{T} \tag{3.26}$$

MATLAB code available at [76] was customized for using in work presented in this thesis.

3.6 MULTICLASS CLASSIFICATION

In the present work, fault classification is a multiclass classification problem, as there are ten types of faults and every fault signal belongs to only one out of them i.e. it is a ten-class classification problem. However, every SLFN has some limitation with their multiclass classification approaches. Therefore, these multiclass classification approaches are avoided by adopting a coding system (Table 5.1) by forming four classifiers (discussed in Section 5.3). However, to understand the benefit of adopting this coding system, this section discusses different approaches of SLFNs for multiclass classification.

3.6.1 SVM Multiclass Classification

SVM is intrinsically a two-class (binary) classifier. To convert it into multiclass classifier following two approaches are prevalent:

3.6.1.1 One-versus-the-rest (OVR) or One-versus-all (OVA)

In this approach K classifiers are created where every model treats one class as positive and all other classes as negative, thus creating total K classifiers, here K is the total number of classes. Class corresponding to the classifier, which has given maximum value of output given by (3.7) is considered as output class. Although this approach is implemented practically, it can give inconsistent results. Moreover, scales of output for different classifiers are most likely to be different.

3.6.1.2 One-versus-one (OVO)

This approach creates classifiers by making pairs of two classes. Thus it forms total K(K-1)/2 classifiers (${}^nC_2$). Output class is decided by maximum votes received by a classifier. This approach can also lead to inconsistent results. Moreover, the numbers of classifiers are also increased. However, in practice, this approach is more popular than one-versus-the-rest approach [77, 78].

3.6.2 RVM Multiclass Classification

Tipping [44, 45] has approached for RVM-multiclass-classifier by formulating generalized form of likelihood given in (3.21),

$$P(t \mid \mathbf{w}) = \prod_{n=1}^{N} \prod_{k=1}^{K} \rho\{y_k(\mathbf{x}_n; \mathbf{w}_k)\}^{t_{nk}} \tag{3.27}$$

where, each class has its own output node (output y_k) and corresponding weights ($\mathbf{w}_k$), only one output remains one and others become zero.

Although this approach is more principled than one-versus-one and one-versus-all, during training this approach is computationally very demanding and infeasible to implement on practical machines. Therefore, for multiclass classification with RVM recourse to one-versus-one or one-versus-all may be required.

3.6.3 ELM and KELM Multiclass Classification

ELM and KELM adopt same approach for multiclass classification. Multiple output nodes are created and corresponding weights are learned analytically. Class related to the output of the node, which has given maximum value is considered as predicted class of input.

Unlike RVM, this approach is very fast and computationally implementable, because the training of ELM and KELM is very fast and learning of weights corresponding to one output node is independent of learning of weights corresponding to other output nodes.

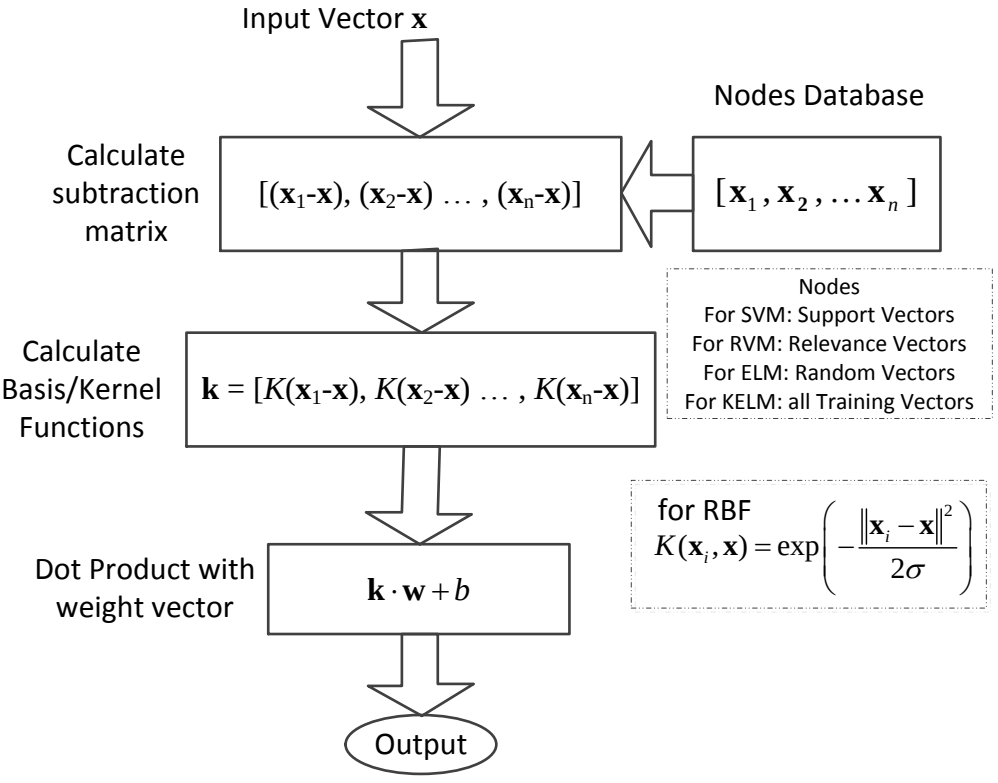


Fig. 3.4: Steps for calculating output of SLFNs.

3.7 FACTORS AFFECTING PREDICTION OF SLFNS

Once all the four SLFNs (SVM, RVM, ELM, and KELM) are trained, they take the form of the following function:

$$y(\mathbf{x}) = \sum_{i=1}^{n} w_i K(\mathbf{x}_i, \mathbf{x}) + b \tag{3.28}$$

where, $\mathbf{x}$ is the input vector during testing (prediction), $\mathbf{x}_i$ are parameters of hidden nodes decided during training, w_i are weights to the output of the hidden layer nodes, b is bias, n is the number of hidden nodes, K is kernel function.

From (3.28), steps involved in calculating the output of an SLFN can be represented as shown in Fig. 3.4.

3.7.1 Kernel Function

Kernel Function is a nonlinear function, which directly gives dot product of two vectors after mapping them in higher dimension space, without calculating projection of vector in higher dimension space. Kernel function can be given by:

$$K(\mathbf{x}_i, \mathbf{x}) = \boldsymbol{\varphi}(\mathbf{x}_i).\boldsymbol{\varphi}(\mathbf{x}) \tag{3.29}$$

where $\boldsymbol{\varphi}$ is a vector projection in higher dimension space after nonlinear transformation.

For SVM and KELM kernel function should satisfy the Mercer's theorem [50, 70, 79]. However, RVM and ELM have no such restriction and work as a simple basis function. Selection of proper kernel function is important for good performance of SLFN.

Radial basis function (RBF) is one of many kernel functions that satisfy Mercer's theorem. RBF is a popular due to its excellent capability of mapping nonlinearities. Moreover, unlike other functions such as polynomial kernel function, RBF has only one parameter. Therefore, RBF requires lesser efforts in parameter tuning. Additionally, during our experiments RBF has given (almost consistently) better performance than other popular kernel functions. Consequently, RBF was used as the kernel function in the present work. RBF is given by:

$$K(\mathbf{x}, \mathbf{y}) = \exp\left(-\frac{\|\mathbf{x} - \mathbf{y}\|^2}{2\sigma} \right) \tag{3.30}$$

where σ is basis width of RBF.

3.7.2 Number of Nodes (Storage and Calculation Time Requirements)

For real-time application of power system protection such as fault classification and section identification, smaller storage memory and calculation time is highly desirable.

From Fig. 3.4 it can be observed that, as the number of nodes increases, the size of nodes database will increase, consequently size of memory required to store this database will also increase. Similarly, with increase in number of nodes calculation time is also expected to increase, because size of the data upon which similar operations have to be performed will increase.

For SVM and RVM nodes are learned during training and they are a subset of the training data. Nodes of SVM and RVM are known as Support Vectors (SVs) and Relevance Vectors (RVs) respectively. The complete training set is taken as nodes for KELM. However, for ELM nodes are decided randomly. Among the four SLFNs, numbers of nodes for RVM are smallest and therefore it is fastest and it has smallest storage requirement, while KELM is slowest and has largest storage requirement.

3.8 Parameter Tuning

The accuracy of an SLFN heavily depends upon the proper selection of values of SLFN's parameters. Normally, due to the issue of over-fitting the performance on the training set is not a good measure of performance on new data. Traditionally, there are two methods for tuning parameters, namely validation and cross-validation.

3.8.1 Validation

If data are sufficient, then available data are divided into three disjoint sets that is training set, validation set, and testing set (Fig. 3.5). Then different models are created by training on the training set through taking different values of parameters. Then the predictive performance of trained models is compared on validation set. The trained model (parameter values), which has given best predictive performance is selected. Sometimes training and validating again and again on same training and validation set may cause overfitting. Therefore, the performance of the selected model is verified on unseen data set named testing set.

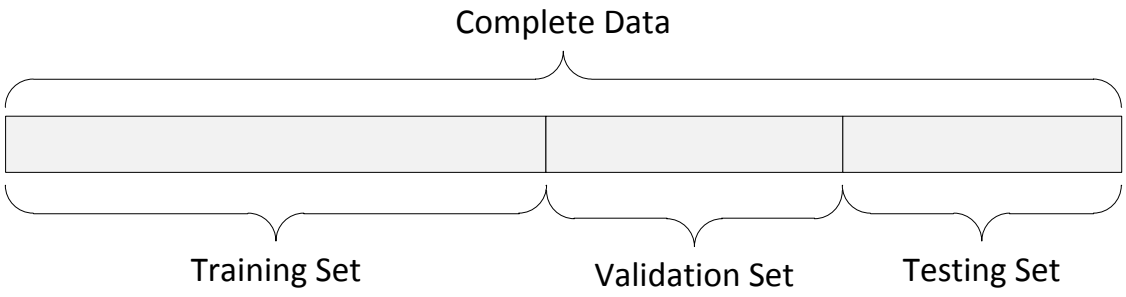


Fig. 3.5: Division of data for validation.

3.8.2 Cross-validation (CV)

However, similar to power system applications, if available data is not sufficiently large then the validation method may prove wastage of data. To prevent it, if size of training set increase and validation set is small then there will be large variations in predictive performance of validation set. Therefore, method of cross-validation (CV) is used for deciding best values of parameters i.e. best model.

For cross-validation, initially data are divided into two disjoint sets named training set and testing set, then training set is further divided into k disjoint subsets (known as k-fold

cross-validation). Out of these k subsets, an *estimation subset* is formed by taking 'k-1' subset together. Then the model is learned from estimations subset with particular values of parameters and predictive performance is verified on a remaining k^{th} subset (called *validation subset*). This process is repeated k times by forming pairs of estimation and validation subsets. Average of predictive performance achieved on k validation subsets is called cross-validation performance (accuracy or error). This process is illustrated in Fig. 3.6 with 4-fold cross-validation. Parameter values that give best cross-validation performance are selected for training complete training set. The performance of the model thus achieved is verified on the testing set.

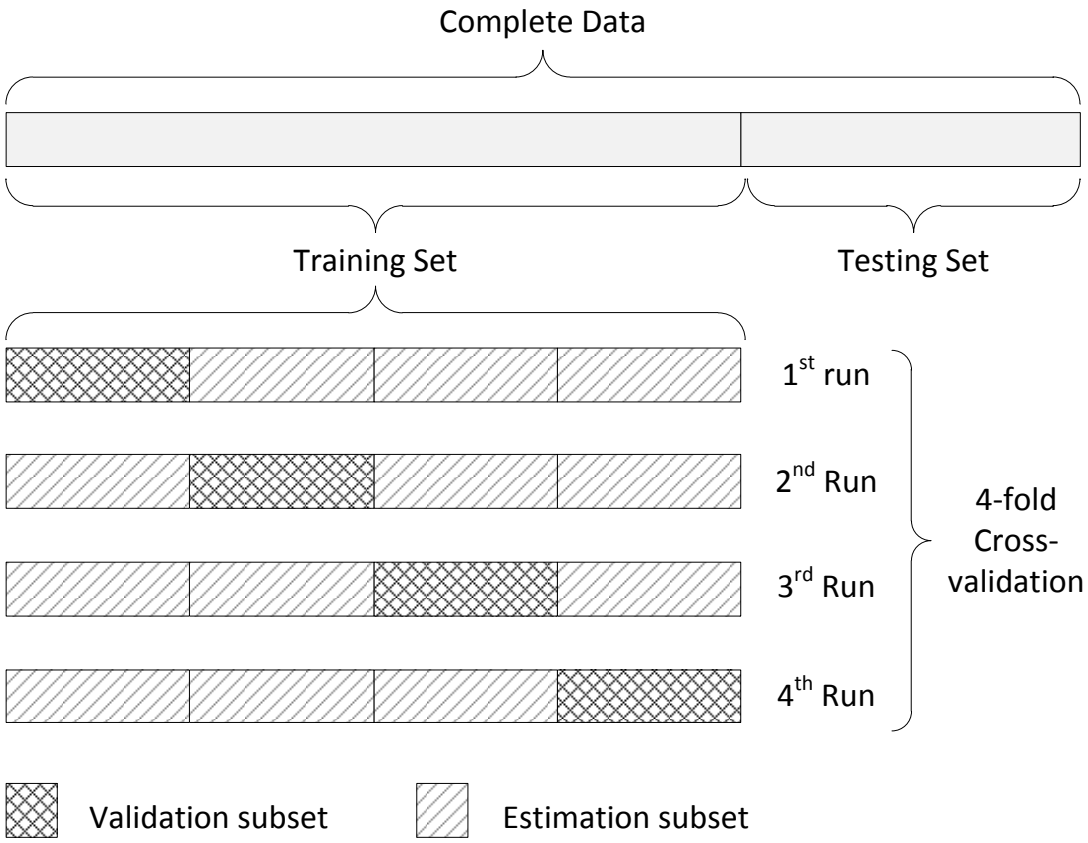


Fig. 3.6: Four-fold cross-validation.

When data are very scarce, then the number of folds during cross-validation may be set equal to the number of elements in the training set, this is termed as *leave-one-out* technique.

One main problem with cross-validation is that the number of training runs increase k times (for k-fold cross-validation), and this becomes even more problematic training time itself is large e.g. in our work training of RVM is itself computationally intensive. Cross-validation becomes further demanding when multiple parameters are to be adjusted. For example, SVM regression has regularization parameter (C), insensitivity parameter (ϵ), and kernel parameters. In the worst case, searching combinations of values of parameters could need number of training runs that are in order of number of parameters.

3.8.3 Ensemble Formation

For SVM, RVM, and KELM, nodes and their number are decided during training procedure. Moreover, their parameters are tuned using cross-validation. However, ELM does not require parameter tuning and hidden nodes are assigned randomly. Moreover, ELM does not restore to validation and cross-validation. Although this saves a huge amount of time, a large number of nodes are chosen to make ELM capable of mapping high degree of nonlinearity (in this work 600 to 2000 nodes). Nonetheless, when weights of large number of randomly assigned nodes are decided analytically using the pseudo-inverse, which may cause over fitting on training data [80]. Therefore, ELM can give poor performance on unseen data i.e. very noisy output. To improve performance of ELM, variation in outputs of different ELM models is averaged out by forming ensembles of many ELM models. Work in present thesis used ensemble of 100 ELMs everywhere. In case of regression, such as fault location, a simple average of outputs of every ELM model of ensemble is taken. While for classification (section identification), the class which has received maximum votes is considered as predicted class.

3.9 GENERAL COMPARISON OF SLFNS

Although detailed comparison of SVM, RVM, ELM, and KEKM are available in the literature [44, 46-48, 50, 78, 81], here a very general comparison on criteria relevant to this thesis is given.

Unlike conventional ANNs, SVM uses the principle of structural risk minimization i.e. formulates convex optimization problem, thus SVM does not stick in local minima. SVM has fewer parameters to adjust, in contrast with ANNs where many parameters require adjustment such as number of layers, number of nodes in each layer, learning rate etc. However, SVM has training time larger than training time of ELM and KELM (lesser than RVM). This makes tuning of even few parameters difficult.

Because of the Bayesian framework RVM avoids overfitting and model formed is very sparse i.e. has very few nodes. Due to the smaller number of nodes, testing time and storage requirement of RVM are very small. This makes RVM particularly interesting for real-time applications such as protection of series compensated line. It has no intrinsic parameter to adjust, only kernel (basis) parameters require adjustment. Since the kernel function is simply a basis (mapping) function, it does not require to fulfill Mercer's condition. However, training time of RVM is largest among all four SLFNs. Consequently, training and parameter tuning of RVM is demanding.

In ELM, nodes of the hidden layer are assigned randomly, only weights to the output of hidden layer are learned by using Moore–Penrose generalized inverse. Therefore, training time of ELM is very small (compared to other SLFNs). Moreover, ELM does not require

parameter tuning, therefore it saves from the validation or the cross-validation. However, to avoid overfitting formation of ensemble is required, which increases ELMs testing time.

Training time of KELM is usually very small (more than ELM) in comparison with SVM and RVM. Moreover, KELM achieves accuracy better than ELM, SVM, and RVM (will be shown in Chapter 1: and Chapter 1:). However, due to the large size of the network, its testing (prediction) time is largest. This makes it attractive for non-real-time applications such as fault location, where prediction accuracy is more important than prediction time. Nevertheless, prediction time is lesser than application requirement (fraction of one second).

From discussions of the present chapter, a comparative summary can be drawn as shown in Table 3.1. Comparisons made in this table are relevant to work in the thesis. However, they are general and non-exhaustive.

Table 3.1: Comparative summary of SLFNs.

	Training Time	**Testing Time**	**Number of Hidden Nodes**	**Parameter Tuning**	**Overfitting**
SVM	Large	Large	Large	C and Kernel Parameters	May Be
RVM	Very Large	Very Small	Very Small	Only Kernel Parameters	Very Unlikely
ELM	Very Small	Small	Small	Not Required	Likely
KELM	Small	Large	= Number of training instances	C and Kernel Parameters	May Be

3.10 CONCLUSIONS

This chapter has introduced the theories of four SLFNs namely SVM, RVM, ELM, and KELM. Although the architectures of all four SLFNs are similar, the size of the learned models and their training procedures are completely different. Depending upon the procedure for training and size of model, time taken for training and testing is also different. Four SLFNs discussed in this chapter have different and sometimes opposite performances and advantages. Therefore, suitability of any SLFN is application specific. Better understanding of the SLFNs will help in deciding the suitability of the algorithm for a particular application.

CHAPTER 4: DATA GENERATION AND SIMULATION STUDIES

This chapter discusses the details of power systems models used for simulation studies. It starts by giving details of two power system models. The first model is a two-area equivalent of power system with an FC and a TCSC at midpoint of transmission line. The second is a 12-Bus 4-Generator model, where FC and TCSC are installed at a midpoint of 600 km long line. Then chapter gives variations of parameters to create scenarios for generating data for training and testing. Finally, chapter proposes a new method for generating data by automatically varying values of parameters. This method has reduced time and efforts to simulate large number of scenarios. Moreover, this method can find many other applications.

4.1 INTRODUCTION

Similar to other machine learning algorithms, single layer feedforward networks (SLFNs) require large amount of data for training, validation, and testing of algorithms. These data should be representative of the possible scenarios that algorithm could face. Fault current and voltage depend on many factors such as:

i) Prefault loading conditions
ii) Compensation level of transmission line
iii) Fault location
iv) Fault type
v) Fault resistance
vi) Fault inception angle

Different combinations of these factors can create numerous scenarios. Preferably, these scenarios should be from actual fault events from the field. However, sufficient fault data is not available from actual events. Moreover, creating faults on the actual power system is not feasible. Therefore, as a normal practice, power system studies take recourse to digital power system simulations for generating training, validation, and testing data.

Therefore, efficacies of the proposed methods were validated on two digitally simulated power systems. These test systems were modeled using PSCAD/EMTDC [52, 82] (generally referred as PSCAD). PSCAD is an industry-standard-software well received by power system researchers and academia. It comes with a library of pre-programmed and tested models of many power system components. Nevertheless, PSCAD has facility of building new custom-components. Inbuilt components were utilized to model two power systems. However, not all needs could be met using inbuilt library components. Therefore, custom-made components were designed to meet the requirements. Different components of power system were modeled in detail such as Transmission Line, Current transformer (CT), and Capacitor Voltage Transformer (CVT).

In our work, fault data were generated by simulating same system model repeatedly by changing the value of system parameters. To facilitate this most of the Electromagnetic

Transient Programs (generally referred as emtp-type programs), including PSCAD, can run the same simulation model in batch mode [83, 84]. Batch mode in PSCAD is named as 'Multiple-Run'. However, batch mode could not change values of some parameters e.g. fault location (transmission line length), source impedance etc. Manual efforts and time are required to change value of these parameters. To save manual efforts this chapter proposes two methods. First method automates the process of changing the transmission line length. However, it cannot change the remaining parameters. To overcome this challenge, a second method is proposed where the value of any parameter can be changed and emtp-type program can be evoked externally. Efficacy of first method was verified by simulating two-area system, while second method was used to generate data for our purpose.

4.2 TWO-AREA-EQUIVALENT POWER SYSTEM

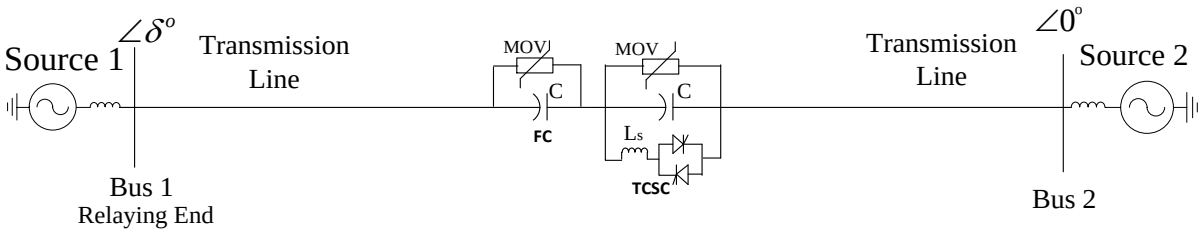


Fig. 4.1: Single Line diagram of two-area-equivalent of power system.

A power system with two areas joined by 300 km long transmission line was simulated in PSCAD, as shown in Fig. 4.1. Two-areas of power system are represented by their Thevenins' equivalent. The transmission line is modeled using frequency dependent (phase) model, which is considered to be the most accurate distributed RLC traveling wave model [85]. Appendix - A (A1) gives details of transmission line parameters and other components of this power system.

To reduce the installation cost and increase the reliability of compensation [53], full compensation is divided into fixed capacitor (FC) and TCSC. FC and TCSC are installed at midpoint of transmission line. Fixed compensation of 25% is provided by FC and a variable compensation from 5% to 15% is provided by TCSC. To provide this variation, firing angle of TCSC is changed between 140º to 180°. Thus, total compensation is varied from 30% to 40%. Compensation level of TCSC is controlled by firing thyristors at appropriate firing angles. Current of transmission line is given to Phase Locked Loop (PLL) for generating a sawtooth synchronization signal that is compared with reference firing angle for generating firing pulses (Fig. 4.2).

Overvoltage protection to FC and TCSC is given by their separate Metal Oxide Varistors (MOVs). In our work, post-fault half cycle data is considered. During this period, energy dissipated in MOV does not go beyond its permitted limit. Therefore, after fault air-

gap does not trigger in the first half cycle. Normally, air-gap triggers after three of four cycles of large fault current. Therefore, air-gap is not modeled here. Ratings of MOV are decided such that MOVs do not start conducting until 1.1 per unit of maximum possible load current.

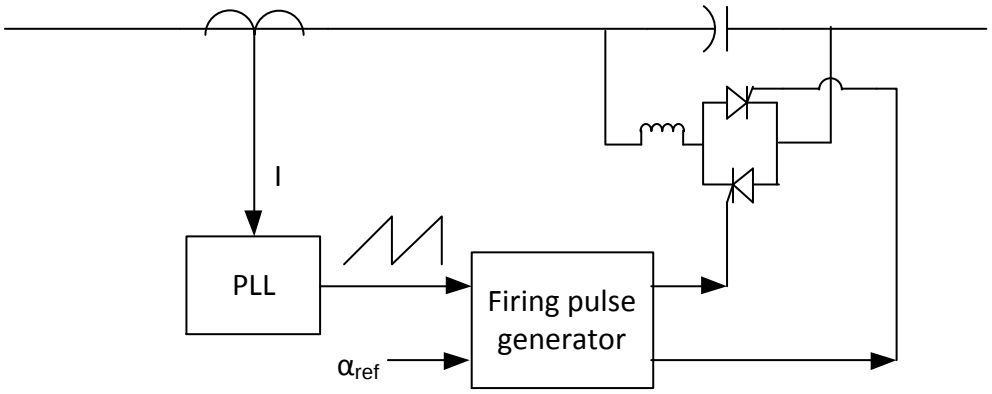


Fig. 4.2: Basic firing control scheme.

Current and voltage signals are measured from current transformer (CT) and capacitive voltage transformer (CVT). CT and CVT are modeled in detail by using Lucas model [52]. Relaying end bus (Bus-1) is installed with CT and CVT. Before sampling current and voltage, signals from CT and CVT were passed through a fourth-order Butterworth filter (anti-aliasing filter). Conforming to the Nyquist–Shannon sampling theorem, cutoff-frequencies of anti-aliasing filters were taken as half of the sampling rate of the signal.

Data required for training, testing, and validation, are generated for various scenarios by taking different combination of parameters. Parameters and their variation for simulating different scenarios are given in Table 4.1. On both sides of TCSC, 720 scenarios were simulated for each type of fault (10 types) by randomly choosing values of parameter (Monte-Carlo-Type simulation) in the range given in Table 4.1. Thus, total 14,400 unique scenarios (2-sides × 720-scenarios × 10-types) were simulated. Choices of scenarios for training, testing, and validation will be described in sections where the method for fault classification, section identification, and fault location are described.

Table 4.1: Variation of parameters for two-area-system.

S. No.	Parameter	Values or range of parameter
1	Fault Types (10 types)	ag, bg, cg, ab, bc, ca, abg, bcg, cag, abc/abc-g
2	Fault inception angle	0° to 360°
3	Firing angle	140° to 180°
4	Fault resistance	0 Ω to 200 Ω
5	Load angle of source	10° to 30° (at Bus-1)
6	Fault Locations	0 to 150 km on both sides of TCSC
7	Source impedance	75% to 125% (for both sources)

PSCAD has an inbuilt facility named as Multiple-Run to create such a large number of scenarios. However, PSCAD does not allow changing source impedance and fault locations. Therefore, these parameters were varied by a new approach discussed in Section 4.4.3.

4.3 12-BUS POWER SYSTEM

Twelve-bus four-generator system given in [86] is a large enough to capture the dynamics of a practical power system, still it is small enough to simulate on an emtp-type program. After Load Flow analysis, it was found that the power flowing through line 7-8 is 330 MW at normal load condition. However, rated power of this transmission line is 500 MW. Therefore, the full capability of line 7-8 was utilized by compensating it i.e. still around 35% can be used after compensation. Therefore, the 12-Bus system given in [86] was modified by installing FC and TCSC in the middle of this 600 km long line 7-8. Fig. 4.3 shows the single line diagram of the 12-Bus system used in this work.

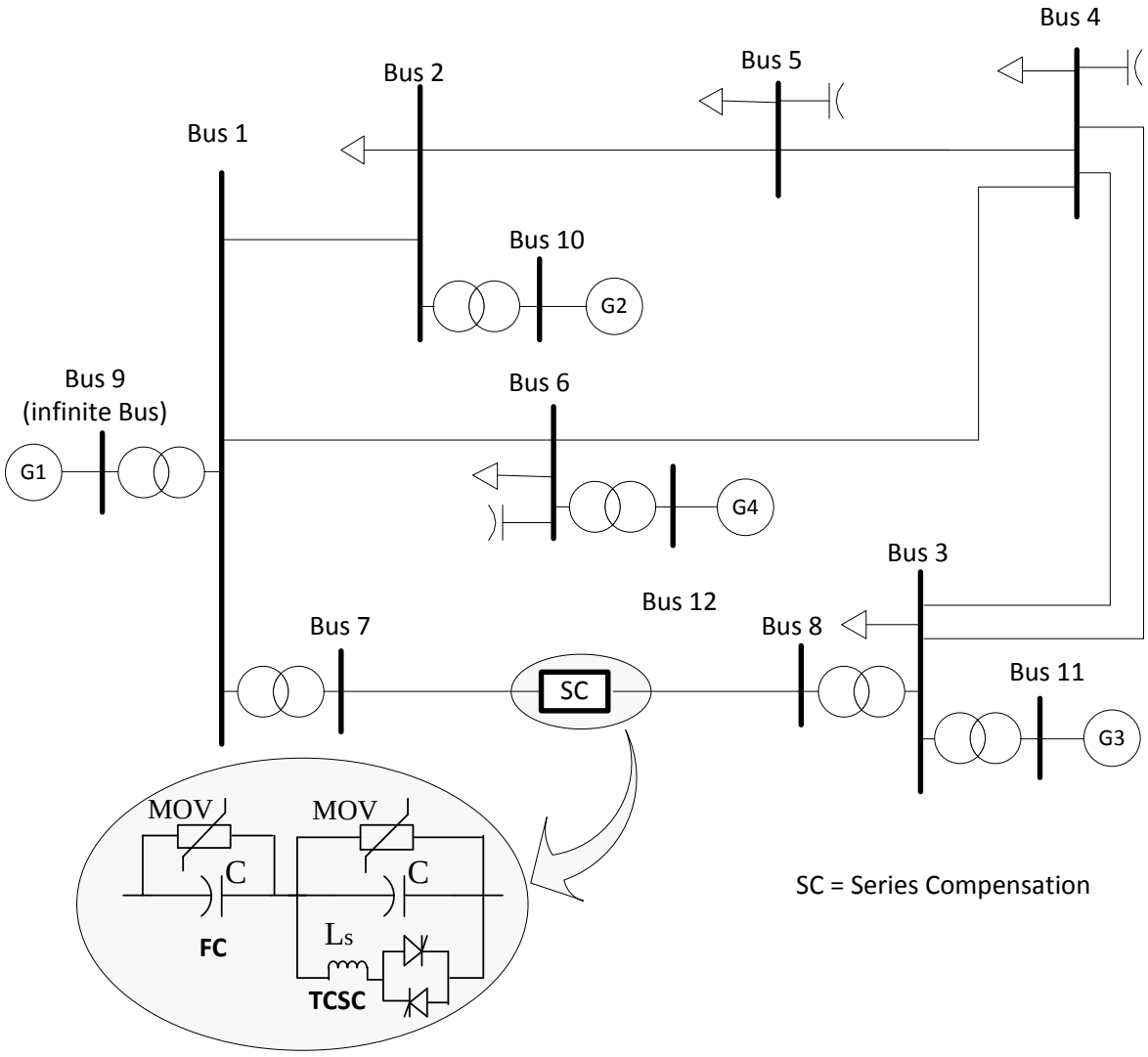


Fig. 4.3: 12-Bus system with series compensated line 7-8.

Similar to two-area system, for economic reasons, compensation was divided into two parts, 20% compensation was provided using Fixed Capacitor (FC) and rest 15% was provided using TCSC at the boost factor of 3 i.e. TCSC-compensation varied from 5% (at firing angle of 180 degree) to 15%. To control compensation, firing pulses were generated using current as a reference to PLL (Fig. 4.2).

The transmission system of this network is 230 kV and 345 kV, except line 7-8 all lines are at 230 kV level. All the transmission lines are modeled in detail using Frequency dependent (phase) model. Generators are also modeled in detail along with their exciters. Moreover, other components such as transformers, loads (active and reactive), and capacitors are modeled in detail. Voltage and current signals were collected from bus-7 considering it as the relaying end of the transmission line.

Performances of protection approaches proposed in this thesis were validated by creating different fault conditions on series compensated line (7-8). Combinations of system and fault parameters taken to simulate different fault scenarios are given in Table 4.2. Out of these parameters, values of first four parameters were varied using inbuilt Multiple-Run facility of PSCAD. Values of remaining parameters were changed with the help of the method proposed in Section 4.4.3.

Table 4.2: Variation of parameters for 12-Bus-system.

S. No.	Parameter	Values or range of parameter
1	Fault Types (10 types)	ag, bg, cg, ab, bc, ca, abg, bcg, cag, abc/abc-g
2	Fault inception angle	0° to 360°
3	Firing angle	140° to 180°
4	Fault resistance	0 Ω to 200 Ω
5	Loads at 5 buses	90% to 110% of nominal load (10 parameters)
6	Load angle at generator buses	Calculated from load flow analysis using other system conditions (3 parameters)
7	Fault Locations	0 to 300 km (2 parameters on each side of TCSC)

Three-hundred scenarios were simulated on both sides of TCSC for each type of fault (10 types). Except fault-type other parameters were chosen randomly (Monte-Carlo-Type simulation) in the range given in Table 4.2. Thus, total 6,000 unique scenarios (2-sides × 300-scenarios × 10-types) were simulated. Choices of scenarios for verifying performance of proposed algorithms will be discussed in detail in corresponding sections of proposed algorithms.

4.4 ENHANCING BATCH MODE CAPABILITIES OF EMTP-TYPE PROGRAM

Previous two sections give combinations of parameters to simulate large number of fault scenarios. This requires simulating same system repeatedly with different combination

of parameters. Therefore, many electromagnetic transients simulation programs (commonly referred as emtp-type programs) [87] can run simulations in batch mode with different values of parameters. However, many values of parameters cannot be changed (treated as constant) as changing them may add additional complexities. Changing some of the parameters values require modification of the complete nodal admittance matrix [87, 88] of a modeled power system. Changing admittance matrix requires external invocation of the simulation engine. Previously in [89], it has been noticed that “the process of externally invoking the simulation engine can require extensive delicate programming, because commercial programs have their own user interfaces and may often lack the documentation outlining the necessary steps to connect to external programs." Thus, manual efforts are required to change the values of these parameters.

4.4.1 The emtp-type Program used for Study (PSCAD/EMTDC)

To understand the problem and the proposed solution, it is pertinent to understand a commercial emtp-type program. The proposed method can be used with any emtp-type program that stores values of parameters in the preformatted text file. Here we have used PSCAD/EMTDC [52, 85] one of the such commercial simulation tools, which comes with the capability of running simulations in batch mode. Batch mode of PSCAD is named as ‘Multiple-Run’.

PSCAD is GUI (Graphical User Interface) of EMTDC. Power system models are simulated in PSCAD by graphically assembling the network [90]. These graphically assembled networks along with the values of parameters entered in it are saved in preformatted text files. Values of parameters are entered in the input fields of the GUI. PSCAD collects all input data from input fields and passes them to the *EMTDC simulation engine*. These input fields are of two data types: real and integer. These data types can be further subdivided into three types: variable, constant and literal.

Values entered in ‘variable input field’ can be changed with the help of Multiple-Run, e.g. fault inception time, fault resistance, breaker switching time, load angle of source, etc. However, values of many parameters of constant and literal type input fields cannot be changed with Multiple-Run, such as transmission line length, source impedance, all parameters of transformers and DC machine, etc. Therefore, to analyze the effect of these parameters, until now, these parameters were changed manually through the GUI of PSCAD. However, manually changing parameters is a time-consuming task and requires lots of human efforts. This becomes a daunting task when values of combinations of more than one parameter are required to change. Sometimes changing the values of parameters requires a variation of values within a particular range according to specific random distribution (Monte-Carlo-type simulation). Doing it manually is complicated and strenuous. Moreover, it is prone to error.

To overcome the above-mentioned problems two methods are proposed in the next two subsections.

4.4.2 Changing Transmission Line Length Parameter Automatically

This subsection proposes a method to change the length of the transmission line automatically. This finally helps in automatically changing location of fault. To understand the proposed method it is essential to understand the general program structure of PSCAD/EMTDC which is shown in Fig. 4.4 [85]

4.4.2.1 PDCAD/EMTDC program structure

PSCAD is a Graphical User Interface (GUI) of EMTDC. Most of the information entered in PSCAD GUI is passed to the EMTDC solution engine directly. However, entered values of transmission line length in the transmission line GUI are collected in preformatted .tli (transmission line input) file. Then this .tli file is given as input to the Line Constant Program (LCP) which is separately available as an executable file (tline.exe). LCP generates a preformatted .tlo (transmission line output) file that contains detailed data of transmission line to help EMTDC to construct a two-port network equivalent of transmission lines for the system. This .tlo file is read by EMTDC during initialization of each run of Multiple-Run as shown in Fig. 4.4.

PSCAD/EMTDC allows user to create his own custom components, for which FORTRAN code or MATLAB interface can be added in DSDYN or DSOUT sections (Fig. 4.4).

4.4.2.2 Proposed methodology to change transmission line length

Proposed method enhances the capabilities of PSCAD/EMTDC by modifying its program structure. This modification makes possible, change of transmission line fault location automatically in each run. The modified program structure of PSCAD/EMTDC is shown in Fig. 4.5. As discussed in previous section .tlo file is read during initialization of each run. But it cannot be modified during the initialization. Therefore, it is modified in DSDYN section where PSCAD allows user to insert custom code. This custom code modifies .tli file with new value of transmission line length and runs LCP with modified .tli to generate modified .tlo file that will be read by EMTDC during initialization of next run. Therefore, in next run the new value of transmission line length will be considered for simulation. Dotted line is used in Fig. 4.5 to emphasize that although .tlo file is modified in DSDYN but modified .tlo file is read during initialization of next run.

Input file (.tli file) is a preformatted text file. Therefore, this file can be modified by any automatic text editor. PSCAD accepts code written only in MATLAB or FORTRAN language.

Since solution engine of PSCAD (i.e. EMTDC) is written in FORTRAN itself, FORTRAN is efficient to modify .tli file. The proposed method is verified by using MATLAB and FORTRAN both.

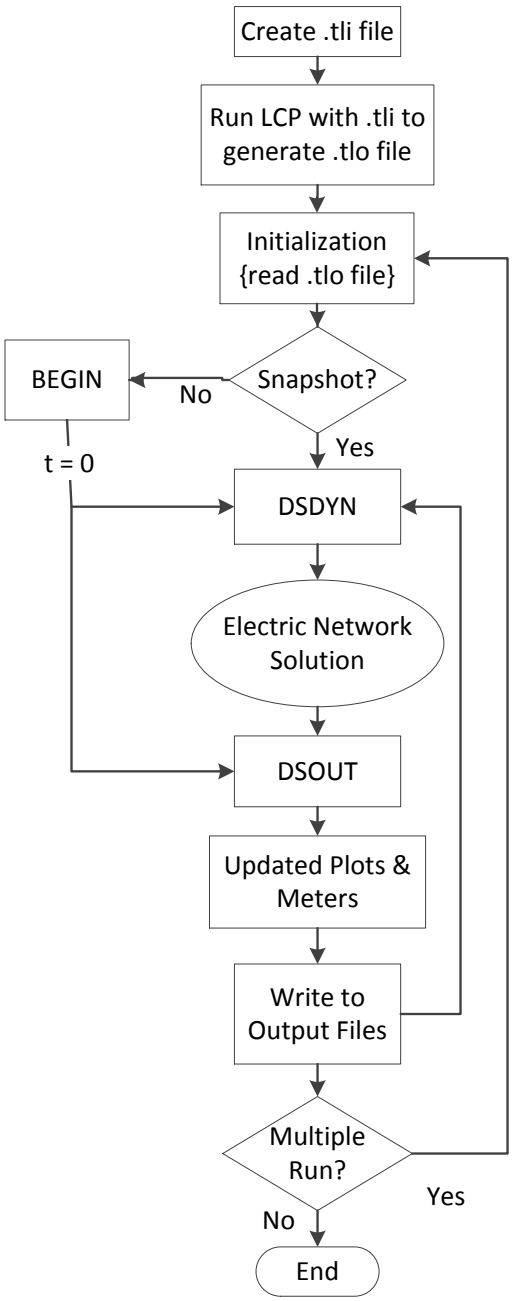


Fig. 4.4: Program structure of PSCAD/EMTDC.

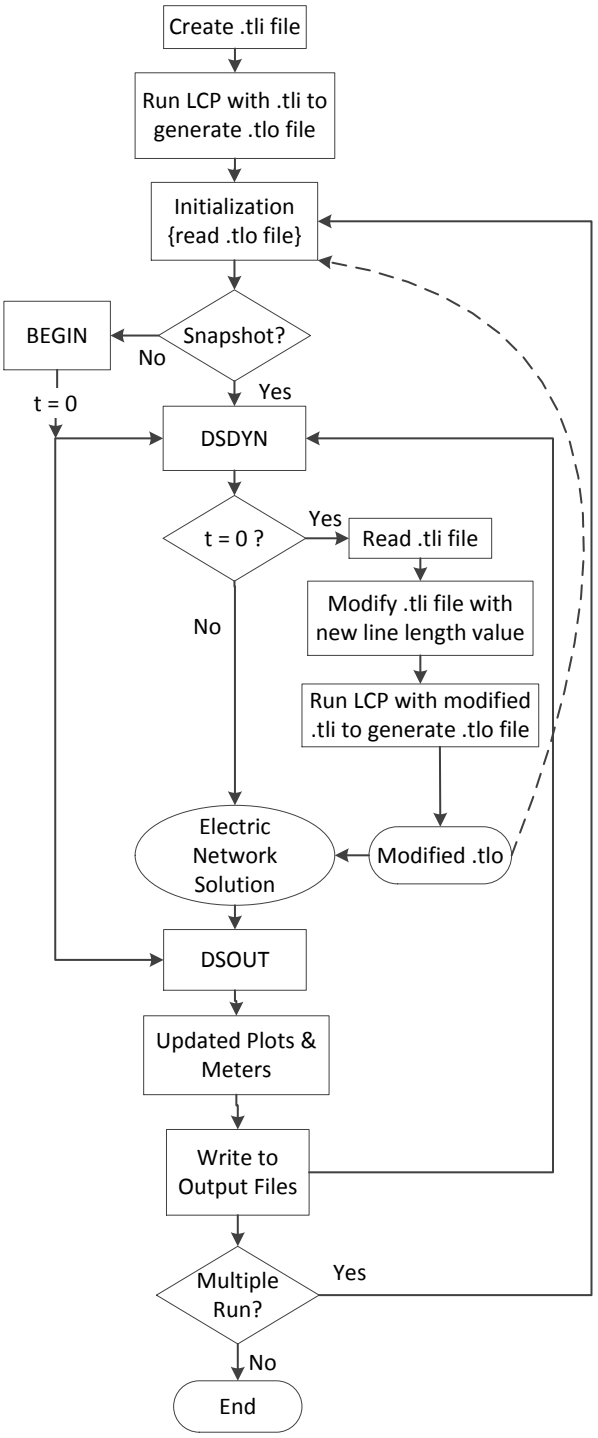


Fig. 4.5: Modified program structure of PSCAD/EMTDC.

For changing transmission line length the .tli file is read line by line. The line that contains “Line Length =” string, is replaced with the new line having a new value of line length parameter.

There are two loops in the program structure of PSCAD/EMTDC (Fig. 4.4), where inner loop is for incrementing the time step of simulation during each run and outer loop is for Multiple-Run. Writing code is allowed in inner loop only. Hence, line parameter is modified in the inner loop, otherwise it would have been more efficient to modify this parameter in the outer loop. Because it would have not been required to check 'if' condition in every step ('t = 0?' condition in Fig. 4.5). This check (t = 0?) is necessary to execute extra code only once in every run i.e. at t = 0, otherwise it will run in every time step of simulation, making the simulation unnecessarily and extremely slow.

4.4.2.3 Simulating different scenarios: Two-area system

To show the significance of the proposed method one case is studied that involved generation of large amounts of data for two area system (Fig. 4.1). Many fault scenarios were simulated with combination of parameters shown in Table 4.1

Data was generated for total 20,000 (10×5×1×1×2×20×10) unique combinations of system parameters (Table 4.1). Values of first four parameters were changed using inbuilt Multiple-Run. Value of source impedance was varied manually and fault location was changed using the proposed method. All the parameter values have been chosen randomly within the above-specified range.

If the proposed method has not been used it would require manually entering values of fault location and source impedance 200 times, where every time 6 values are required to change. These values are positive, zero sequence impedance of both voltage sources (4 values) and transmission line length for both sections of the faulted segment of transmission line (2 values). While using the proposed method, it required entering values of source impedance 10 times, every time 4 values were changed. Therefore, without using this method 1200 values would have been entered manually. However, using proposed method, only 40 values were entered manually.

Similar data were generated for transmission line fault classification in [91] where the fault location was changed manually and the data generation took nearly two person-weeks. However, this time the proposed method reduced data generation time to two person-working days and simulation engineer is saved from drudgery of manual task.

4.4.3 Externally Modifying Parameters and Invoking Electromagnetic Transients Simulation

The method proposed in previous subsection could internally change the length of transmission line. However, previous method could not change values of other parameters and still manual efforts are required. To overcome this current subsection proposes a method

that can change the value of any parameter including transmission line length. This method adopts a two-step approach:

1. Externally change the values of parameters.
2. Invoke emtp-type program externally, which in turn invokes simulation engine.

This can be achieved without knowing the user interface and hence saves from “extensive delicate programming." This method reduces time and efforts involved in many complex power system studies. Although proposed method is used for generating fault data by creating different fault scenarios, this method is general in nature. Therefore, it can find many other applications in power system studies such as:

1. Generating current and voltage signals by creating different scenarios that can be replayed later for testing of control or protection devices.
2. Generating current and voltage samples or other processed features for training and testing of artificial neural network (ANN) based algorithms (used in this work).
3. Generating a large number of signals for further analysis.
4. Finding optimum values of parameters based on some performance indices (PI) for designing purpose [89].

Automation of the whole process can save several person-hours. Since commercial emtp-type programs are used extensively in power system industry for design and analysis purpose, saving person-hours results in saving the project and development cost. Two steps of the proposed method are independent of each other and can be used in different combination depending on the application.

The process of changing the values of parameters involves following two steps:

4.4.3.1 Modifying the values of parameters (Step I)

The name of any particular parameter and its value is stored as a preformatted string in a file of emtp-type program. This information is utilized for automatically changing the values. Text string containing the name of the parameter is searched within the file and searched string is replaced with modified string having a new value of parameter.

Any automatic text editor can perform this task. In this work, MATLAB is used for this purpose. Fig. 4.6 explains the detailed procedure.

Before applying the proposed method, simulated power system model should be verified and saved in a file. For the sake of better understanding of the method, the saved file is referred as *Source* and the new file (that will be created with the modified values of the parameters) is referred as *Target* in further discussion. *Target* file can be created using the following steps:

1. Open Source file as text.
2. Create and open new blank Target file.
3. Read Source file line by line.

4. If the line string contains any parameter name (whose value is to be changed), then replace the value of the parameter in the string with the new value and copy this modified string in Target file.
5. If the line string does not contain any parameter name (whose value is to be changed), then copy the string in Target file without modification.
6. Repeat steps 3 to 5 for each line until the end of the Source file is reached.
7. Close both files.

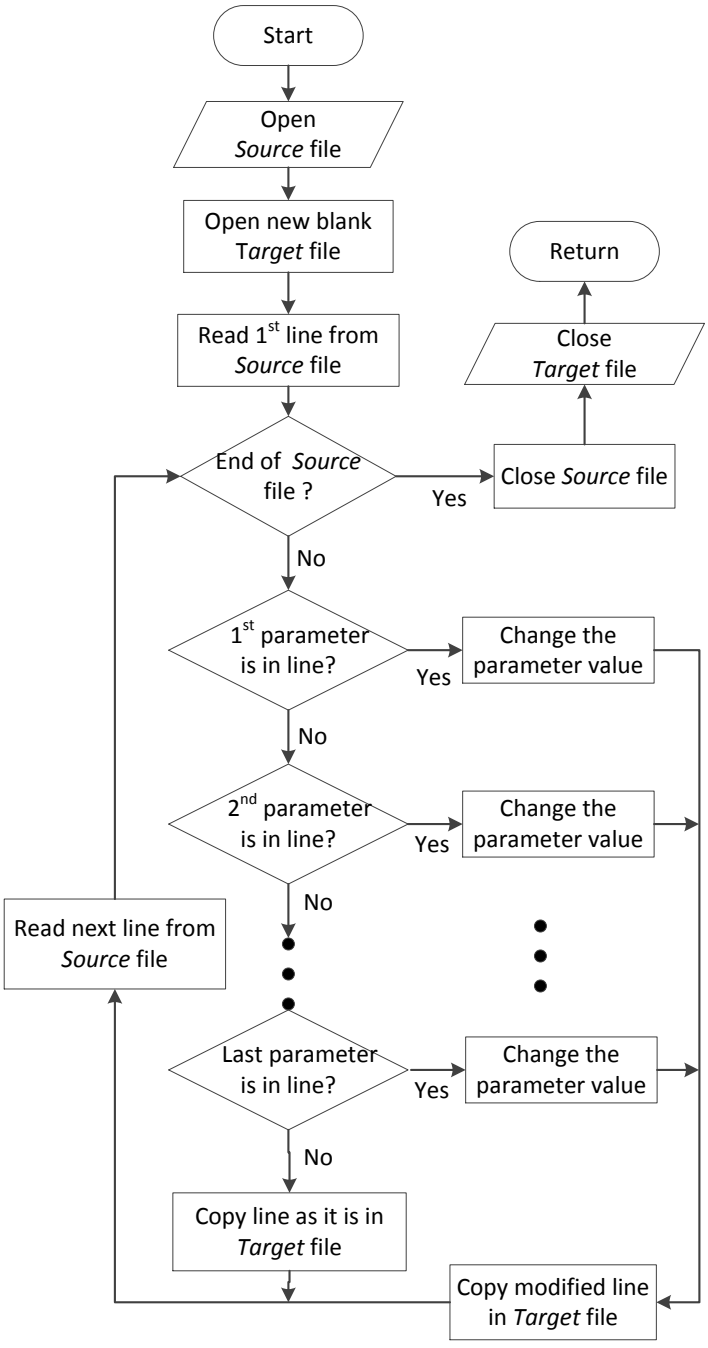


Fig. 4.6: Modifying values of parameters.

4.4.3.2 Externally invoking emtp- type program (Step II)

After generating file automatically, it can be opened as usual, but still it will take manual supervision for doing so. Therefore, to automate this step any software, which simulates keystrokes, mouse movement and window control and manipulation, can be used. AutoIt [92] and AutoHotkey [93] are few of the freeware available for the task. To invoke emtp-type program for running *Target* file, one batch file can be written. This file contains keystrokes and window control-and-manipulation commands such as:

1. Activate Window of emtp-type program.
2. Open *Target* file.
3. Run the simulation.
4. Wait for completing the run.
5. Unload the file.

4.4.3.3 Applications of the method

Two steps of the proposed method discussed in previous subsection are independent of each other. Therefore, they can be used in different combination depending on the power system application. This subsection describes procedures of applying proposed steps in two different applications in two distinct ways.

The proposed method is very general. Therefore, it can be utilized in many other applications as per requirement. It is general in the sense that values of any parameter can be changed automatically, including the simulation time step. Moreover, values can be varied according to previously available data or sequentially or even randomly (Monte-Carlo-type). In addition, method can be used in combination with inbuilt Multiple-Run facility.

(a) Generating signals for different operating conditions

Some power system applications (including present work) require collection of voltage and current signals associated with different values of system parameters and operating conditions for further analysis. One of the ways of using the proposed method for such applications is given in Fig. 4.7. Initially create all the *Target* files with modified values of parameters in one shot and then invoke the emtp-type program to run all the files one by one.

(b) Searching optimal values of parameters

Applications of design require a search of optimal values of parameters based on some performance indices (PI) [89, 94].

Procedure of searching optimal values is shown in Fig. 4.8. Create *Target* file with initial values of parameters given by the search algorithm. Invoke the emtp-type program to

run the file. Calculate PI, if stopping criteria not met, evaluate new values of the parameter by an optimization algorithm. Repeat these steps until stopping criteria met.

The method gives the flexibility of utilizing many optimization packages available. Although some commercial emtp-type programs (e.g. PSCAD) have the facility of inbuilt optimization, they have very limited choice of optimization algorithms [94]. Moreover, the inbuilt optimization facility can optimize only those objective functions, which depend on parameters that can be modified internally, but the proposed method can optimize objective functions dependent on any parameter.

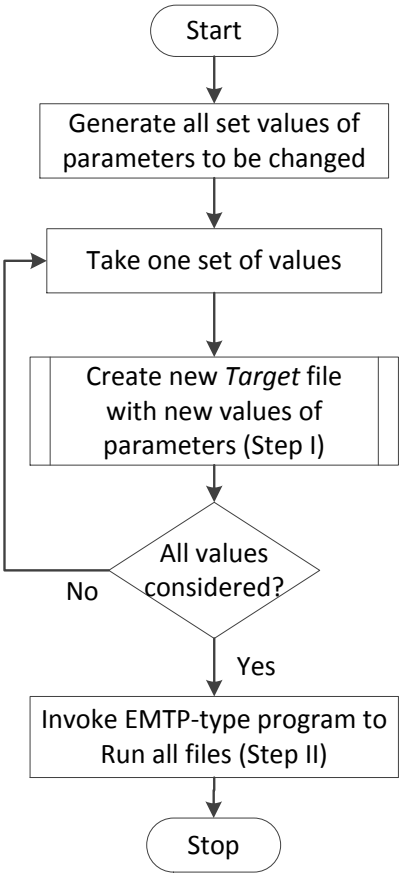


Fig. 4.7**:** Signal generation with different values of parameters.

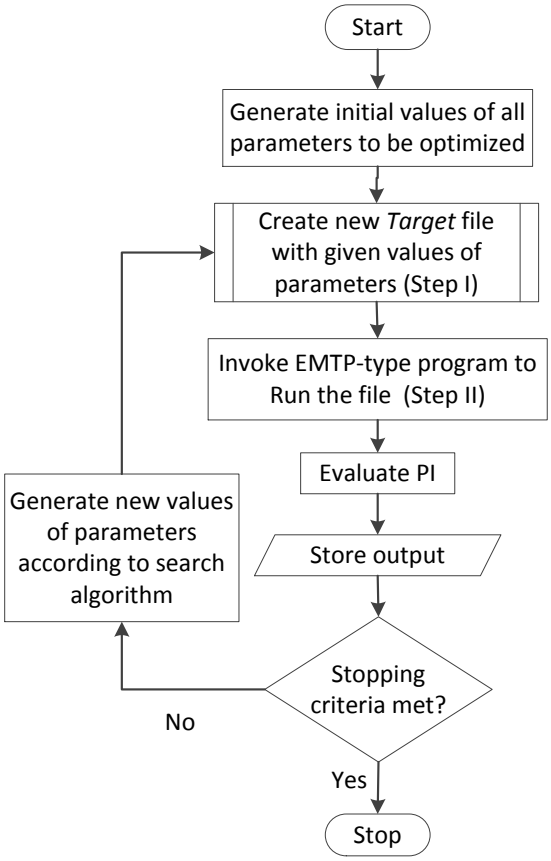


Fig. 4.8: Searching optimal values of parameters.

Next two subsections give details of implementation of the proposed method for simulating many fault cases in the present work.

4.4.3.4 Simulating different scenarios: Two-area system

As discussed in section 4.2, total 14,400 unique scenarios were simulated as per variation of parameters given in Table 4.1. Values of first five parameters in Table 4.1 were changed using inbuilt Multiple-Run facility of PSCAD. However, inbuilt Multiple-Run cannot change values of source impedance and fault location. Therefore, their values were changed using the method proposed in the present section.

Total 144 files for both sides of TCSC were created with different values of fault location and source impedance. These files were created using first step of the proposed method. Then all these files were run using second step of the proposed method. If the proposed method has not been used it would require manually entering values of fault location and source impedance 144 times, where every time 6 values are required to change. These values are positive, zero sequence impedance of both voltage sources (4 values) and transmission line length for both sections of the faulty segment of transmission line (2 values).

Therefore, without using the proposed method, 864 (144×6) values would have been entered manually, nevertheless now all values have been taken automatically. Similar data were generated for fault classification in [95] where these values were changed manually and the data generation took nearly two weeks. However, this time the proposed method reduced data generation time to one working day and simulation engineer was saved from the manual task of nearly two weeks.

4.4.3.5 Simulating different scenarios: 12-Bus system

For 12-Bus system [86] total 6000 unique scenarios were simulated. These scenarios were created by changing parameter according to variation given in Table 4.2. In this case, first four parameters in Table 4.2 were modified by inbuilt Multiple-Run option of PSCAD. However, Multiple-Run could not change the load at five buses, load angle at generator buses, and fault locations.

Total 60 files were created and run with the help of the proposed method. In every file, 15 parameter values were changed externally. Thus, total 900 (60×15) values were modified automatically using the proposed method. This would be a daunting task otherwise. Sometimes the values of few parameters depend on some other parameters. Then calculating values of dependent parameters manually (or using separate program) and entering them into the emtp-type program will require time and efforts. The proposed method can automate this process. In the present case, following were few such occasions:

a) The length of one faulted section of transmission line depends upon the length of the other faulted section, because transmission line length is the sum of the length of both faulted sections. This process was automated using MATLAB.
b) Depending upon load conditions, load angles at generator buses were calculated using optimal power flow (OPF) program MATPOWER [96]. These values were filled using the proposed method. OPF program was seamlessly integrated with the proposed method since code for both is written in MATLAB.
c) Simulation time step depends upon the length of the smallest transmission line segment and some other factors. These factors are checked and if required, simulation time step

is changed accordingly. Using this further time has been saved by selecting larger simulation time step, wherever it was possible.

Simulated system was large and simulation time step was small. Moreover, each file had many inbuilt Multiple-Run simulations. Therefore, complete simulation time for each file was large. Since these files were run automatically, no continuous manual supervision was required for running next simulation after completion of current simulation. This avoids large idle waiting time while simulation is running. This case has taken seven person-working days, without the proposed method it may have taken months to complete the task.

Current and voltage waveforms from relaying end were saved so that any protection algorithm can be analyzed later by replaying these waveforms. This can save a large amount of time, since replaying only stored waveforms will be much faster than running complete simulation. Advantages of the proposed method over manual method are summarized in Table 4.3.

Table 4.3: Summary of advantages of proposed method.

Advantages	**Proposed Method**	**Manual method**
Can change values of	All parameters	Some parameters
Parameter optimization	Yes	No
Monte-Carlo-Simulation	Yes	No
Manual Supervision	No	Yes
Prone to error	No	Yes
Toiling Task	No	Yes
Idle waiting time	No	Yes
Time Saving	Yes	No
Total simulation time Two-area system	1 day	2 weeks
Total simulation time 12-Bus system	1 week	> 1 month

4.5 CONCLUSIONS

This chapter introduced two power system models for verifying the efficacy of methods proposed in this work. These methods are based on learning algorithms that require large data set for training and cross-validation. Therefore, the chapter proposed two novel approaches for simulating a large number of scenarios in batch-mode to record the current and voltage signals of faults. In the first approach, program structure of a popular emtp-type program (PSCAD) is slightly modified to change the transmission line length in batch mode. This resulted in reduced time and efforts for signal generation process to a large extent. However, this method was not able to change the values of many parameters other than transmission line length. Therefore, a generalized method to change the values of any

parameters was proposed by adopting two-step approach. This method further reduced time and efforts for simulating different scenarios in batch mode. Moreover, its generalized nature throws open possibilities of its utilization for other applications such as parameter optimization.

This chapter presents two methods for fault classification, one using SVM and another using RVM. Implementation procedure and results are discussed in detail. Parameters of SVM are tuned with genetic algorithm (GA), which yielded better results. SVM achieved high fault classification accuracy. However, prediction time of SVM is high due to a large number of support vectors. RVM has accuracy comparable to SVM. However, RVMs have smaller storage and calculation requirement than SVM.

5.1 INTRODUCTION

Knowledge of type of fault is essential for protection any transmission line. However, in series compensated lines identifying type of fault is hard due to nonlinearities caused by protection equipment of fixed capacitor (FC) and TCSC. Support Vector Machine (SVM) and Relevance Vector Machine (RVM) are two SLFNs that are widely popular for their capability of mapping nonlinearities. Therefore, this chapter proposed methods using SVM and RVM for identifying type of fault (fault classification). All ten-type of shunt faults are identified, namely ag, bg, cg, ab, bc, ca, abg, bcg, cag, abc/abc-g. Similar to any other machine-learning algorithm, selection of features containing relevant and discriminatory information is vital for SVM and RVM.

Due to the symmetric nature of three-phase faults (abc and abc-g), their zero-sequence currents are negligible and similar to each other. This can be seen in Fig. 5.1 that is a zoomed-out version of Fig. 5.2 (d). Therefore, as a general practice, abc-g faults (with ground) were treated as abc faults (without ground).

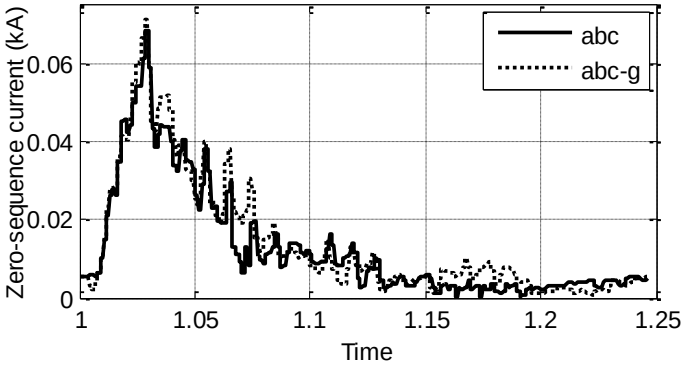


Fig. 5.1: Zero-sequence current of three-phase faults.

The procedure for selecting appropriate parameters for SVM and RVM is also prime concern for obtaining optimal results. Cross-validation was used to tune parameters of SVM and RVM. Based on the input features, factors affecting the fault classification accuracy are discussed with results. Performance of SVM and RVM are compared with each other on the basis of classification accuracy, storage requirements, and calculation time for fault classification. Storage (memory) and calculation time should be small enough to be implemented in real time application i.e. fault classification. RVM gives accuracy comparable

to SVM. Although training time of SVM is lesser than RVM, memory and calculation requirement of SVM are higher than RVM.

5.2 FEATURE SELECTION

"A feature is a characteristic of an observed phenomenon, which can be measured and given as input to a pattern recognition algorithm." In this work, fault-type-identification is treated as a pattern recognition problem, where patterns of faulty signals are identified to classify them to particular fault type. Selection of relevant and discriminatory feature is crucial for good performance of any classifier. Sometimes relevant information is extracted by preprocessing the raw data, so that extracted features can distinguish one class from others. However, if explicit features are unknown, then raw data can be given directly to the classifier. In this case, classifier itself identifies features and patterns from raw data.

Similarly, in our work raw samples of available signals were used for phase classification. Still, selected raw data should possibly contain the 'relevant and discriminatory information'. To select inputs for classifiers that should have 'relevant and discriminatory information', following questions should be addressed:

1. Which signals are to be taken as input, current or voltage?
2. Whether samples from all phases should be given to each phase classifier or samples from individual phase should be given to the corresponding phase (same phase) classifier?
3. What should be the optimal data window length for better accuracy?
4. What should be the sampling frequency of the signal?
5. How many training instances are sufficient for good performance of classifier?

These questions are answered in results and discussions section (Section 5.6) by giving results and offering rationale behind results obtained.

Although raw data were given to phase-classifiers, involvement of ground in fault was detected using zero-sequence current as explicit feature. Ground faults have higher zero-sequence current than faults not involving ground. Therefore, zero-sequence current can offer relevant information that can discriminate between ground and non-ground fault. Fig. 5.2 shows the magnitude of zero-sequence current for different type of faults while all other parameters are same. In Fig. 5.2, ab fault (Fig. 5.2 (a)) has lesser zero-sequence current than ab-g fault (Fig. 5.2 (b)) has. Similarly, c-g fault (Fig. 5.2 (c)) has large value of zero-sequence current. However, abc and abc-g faults (Fig. 5.2 (d)) have negligible zero-sequence current compared to other faults. Thus, zero-sequence current was chosen as a feature for the classifier to discriminated between ground fault and non-ground fault.

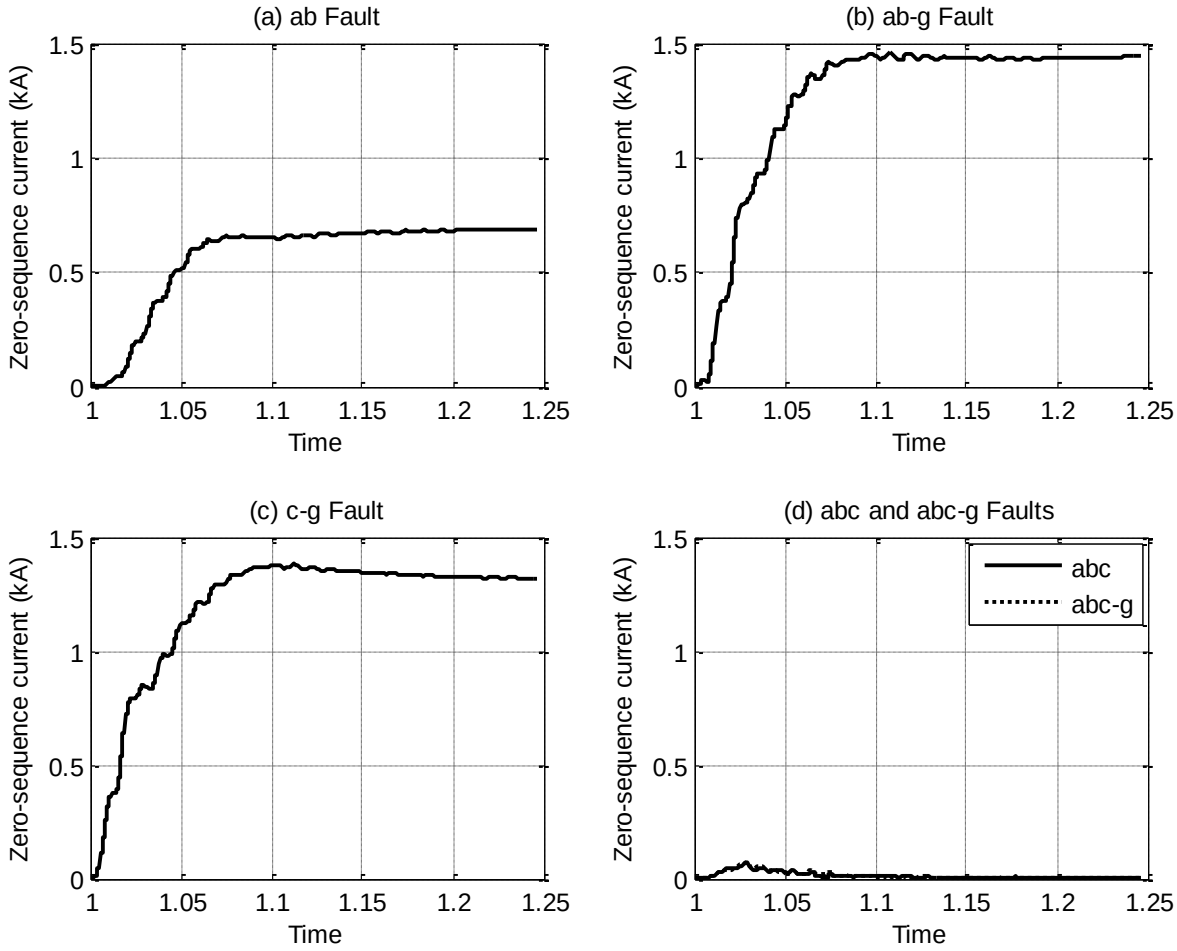


Fig. 5.2: Magnitude of zero-sequence current with ground and non-ground fault (a) ab Faut (b) ab-g Fault (c) c-g fault (d) abc and abc-g fault.

5.3 PROPOSED METHOD FOR FAULT CLASSIFICATION

Based on discussion in previous section, this section proposes a method for fault classification in series compensated line. Scheme of the proposed method is represented diagrammatically in Fig. 5.3. After measuring the current through current transformers (CTs), current signals were passed thorough antialiasing filter (4^{th} order Butterworth filter). Then current signals were sampled at a required sampling rate. To meet the criterion of the sampling theorem, the cutoff frequency of antialiasing filter was kept half of the sampling frequency of current signals.

Post-fault current samples from all the three phases were given to all phase classifiers (SVM/RVM) to identify whether phase is faulty or not. The zero-sequence analyzer is used to extract zero-sequence component of current that was given to a ground classifier to detect involvement of ground in fault.

Fault type was decided by using output of these four classifiers. Here ‘one’ at the output of a classifier denotes involvement of that phase or ground and ‘zero’ at output means that particular phase or ground is healthy. This way all ten types of shunt faults can be classified according to binary coding given in Table 5.1.

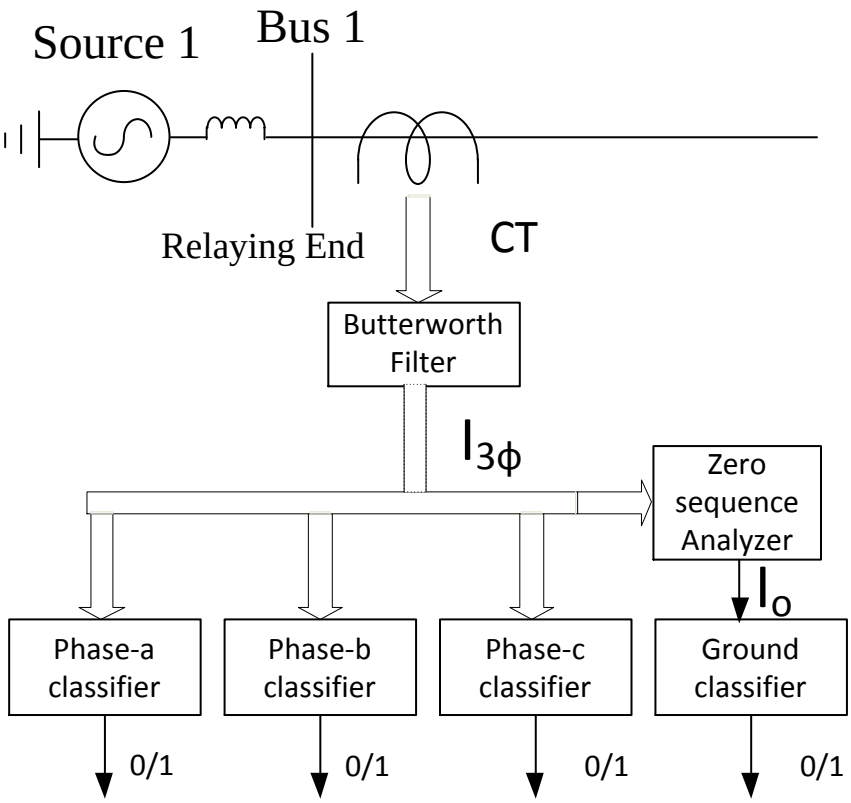


Fig. 5.3: Schematic diagram of proposed fault classification method.

This type of coding is chosen over existing single multiclass (10 class) SVM based technique of [95], because architecture of four classifiers is particularly suitable for RVM and SVM. As RVM and SVM are inherently binary-classifiers, to make them multiclass-classifier one-against-one (OAO) or one-against-all (OAA) techniques are required. These techniques would have internally created 45 ($^{10}C_2$) and 10 classifiers respectively for OAO and OAA. This would have led to extra time for cross-validation, training, and testing.

Table 5.1: Coding for types of faults.

S. No.	a-phase classifier	b-phase classifier	c-phase classifier	Ground	Type of Fault
1	1	0	0	1	a-g
2	0	1	0	1	b-g
3	0	0	1	1	c-g
4	1	1	0	1	ab-g
5	1	0	1	1	ac-g
6	0	1	1	1	bc-g
7	1	1	0	0	ab
8	1	0	1	0	ac
9	0	1	1	0	bc
10	1	1	1	0	abc/abc-g

As a regular practice adopted by other researchers [15, 20, 26-30] it has been assumed that the fault has already been detected by other subroutine before the classification of faults. Classifiers used here are Support Vector Machine (SVM) and Relevance Vector Machine (RVM).

5.4 PARAMETER TUNING

Normally machine-learning algorithms have parameters associated with their training. To achieve good generalization (accuracy on unseen data), the optimal values of these parameters should be obtained before training a learning machine. Traditional ANN algorithms have many parameters to adjust, e.g. number of layers, number of nodes in each layer, learning rate, and parameters related to activation function etc. Interestingly, SVM and RVM have a very small number of parameters to adjust. While SVM has regularization parameter (C) and parameters of kernel-function, RVM has only parameters of kernel function because the regularization parameter is inferred internally during training. After trying different kernel functions, widely popular RBF kernel function was selected for our work. RBF has the additional advantage of having only one parameter (γ).

Five-fold *cross-validation* (CV) was used as measure of searching optimal values of parameters. Cross-validation gives measure to compare two different set of values of parameters (Section 3.8.2). However, Cross-validation does not give a procedure for finding optimal value of parameters. Therefore, values of parameters are generally found through hit and trial or grid search. Nonetheless, these methods may result in suboptimal performance of the classifier. Therefore, use of some optima-searching algorithm seems advisable.

5.4.1 Parameter Tuning of SVM

As a result of the above discussion, Genetic-Algorithm (GA) was used to search optimal values of SVM-parameters. Biological genetic evolution has inspired this global-search technique of GA. It starts in the feasible region by guessing an initial population. The fitness of each member of initial population is calculated by a fitness function (objective function). From an older generation new population is generated by some criteria. Few fittest individuals also survive to the next generation. This procedure is repeated until best individual (here parameter value) of required fitness is not found. In this way, by following the principle of survival of fittest, best individual survive from old generation to the new generation. Therefore, the solution is expected to improve with every new generation. After meeting some stopping criteria, the global optimal solution is represented by fittest individual of last generation.

The SVM-model was cross-validated, trained and tested using libsvm-mat-2.91-1 [71]. The fitness function of maximizing the CV-accuracy was used by Global Optimization Toolbox [97] of MATLAB to search optimal values of SVM-parameters (C and γ).

The fitness function of the GA is to maximize classification accuracy obtained using 5-fold cross-validation is defined as

$$\max\ f\left(C,\gamma\right)=\max\left(\text{CV-accuracy}\right) \tag{5.1}$$

where CV-accuracy is defined as

$$\text{CV-accuracy}=\frac{\text{no. of correctly classified instances in training set using CV}}{\text{total no. of instances in training set}} \tag{5.2}$$

Details of GA used, were as follows:

i) Fitness scaling function: rank
ii) Population size = 20
iii) Selection: Stochastic Uniform
iv) Migration: forward
v) Crossover: Heuristic
vi) Stopping Criteria: Fitness Limit =0.002 and Stall generation limit = 5

The SVM-model was formed by utilizing optimal values of 'C' and 'γ' obtained by the above procedure of GA.

5.4.2 Parameter Tuning of RVM

In case of RVM only parameters associated with basis function (kernel) are required to adjust, as all other parameters associated with RVM are inferred from training data by a Bayesian method within the RVM toolbox. Here RBF kernel is used which has only one parameter i.e. 'γ'. To decide the optimal value of this parameter a simple linear search is performed by cross-validation. The value of the parameter, which has given the maximum cross-validation accuracy, is used for training RVM.

Although, parameters of SVM were searched using genetic algorithm, simple linear search was used to find the value of RVM-parameter. This is because cross-validation-time for RVM is much higher than cross-validation-time of SVM. Moreover, CV-time of RVM increases cubically as the number of training instances increases (n^3 time complexity) [33], in addition GA itself is time taking procedure. Therefore, searching RVM-parameters was not feasible by using GA. Besides, linear search was able to give results comparable to SVM. However, it cannot be denied that better results may be obtained by using GA to search parameter of RVM.

Code of 'Sparsebayes (Version 2.00) [73] MATLAB Toolbox' was used for cross-validating RVM.

5.5 SELECTION OF DATA FOR TRAINING, TESTING AND CROSS-VALIDATION

For two area system, total 14,400 unique scenarios were simulated for cross-validation, training, and testing the proposed algorithm (details in section 4.2 and 4.4.3.4).

Scenarios were simulated by widely varying system and fault conditions such as fault types, fault inception angle, firing angle, fault resistance, load angle of source, fault locations, source impedance. Such a large variation in system and fault condition provides the fault classifier model good prediction capability under real-life scenarios.

Out of total 14,400 instances (scenarios) 4,400 were taken for CV and training and remaining 10,000 were used for testing the classifiers. As a standard practice, training set was kept disjoint from testing set. Testing set disjoint from training set and large number of testing instances with wide variation in scenarios ensure that reported results are free from overfitting.

5.6 RESULTS AND DISCUSSIONS

During discussion of feature selection in Section 5.2, fault classification accuracy was postulated to depend on different factors. The present section (5.6) compares performances of SVM and RVM based on fault classification accuracy. It also discusses factors affecting the fault classification accuracy. Further, the present section verifies effect of these factors by experimental results on simulated data. Extensive experiments were performed on total 792 cases (Fig. 5.4) by taking different combinations of factors that can affect performance of the proposed method. Significantly, discussions in this section set the basis for selection of some parameter settings in the next two chapters.

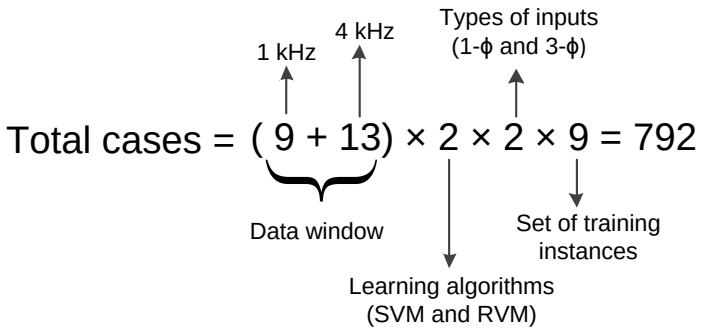


Fig. 5.4: Distribution of total cases studied according to factors affecting performance of the method.

5.6.1 Type of Signal for Phase Selection

Current and voltage samples are the only relevant data available at relaying end. Capacitive voltage transformers (CVTs) are used to measure voltages at extra high voltage (EHV) level. Information contained in high frequency components of voltage signals filter out because of the capacitive nature of CVT, because CVT behaves as a low pass filter. Current transformers (CTs) are used to measure current for relay. Normally, CTs have sufficient bandwidth. Therefore, information content of current signals is expected to be more than voltage signals. Experiments have confirmed this, as the performances of algorithms with current samples were better than voltage samples. Interestingly, results with current samples

alone were better than the combination of current and voltage samples. Therefore, current samples were taken as input for all pattern recognition algorithms used in this work.

5.6.2 Individual-phase Current vs. Three-phase current

Just after fault, distortions in current of faulty phase are more than non-faulty phases (observed from Fig. 5.5). Moreover, magnitudes of current in faulty phases are higher than non-faulty phases (Fig. 5.5). Therefore, current signals (distortions and magnitude) from individual phases should be used to tell whether that phase is faulty or not. However, if distortions and magnitude of current signals from all phases are compared with each other, then identifying faulty phases will be even easier. Therefore, by virtue of relative comparison, current samples from all the three phases given to each phase-classifier (Fig. 5.6 (a)) are expected to give higher accuracy than current samples of individual phase given to corresponding phase-classifier (Fig. 5.6 (b)). This supposition is vindicated by the results shown in Table 5.2 and Fig. 5.7. For the sake of brevity 'current samples from all the three phases given to each phase-classifier (Fig. 5.6 (a))' is represented by '3-ϕ' and 'current samples of individual phase given to corresponding phase-classifier (Fig. 5.6 (b))' is represented by '1-ϕ'.

'3-ϕ' inputs give better results than '1-ϕ' inputs (Table 5.2 and Fig. 5.7) when all other factors (e.g. data window length, classification algorithm) are same.

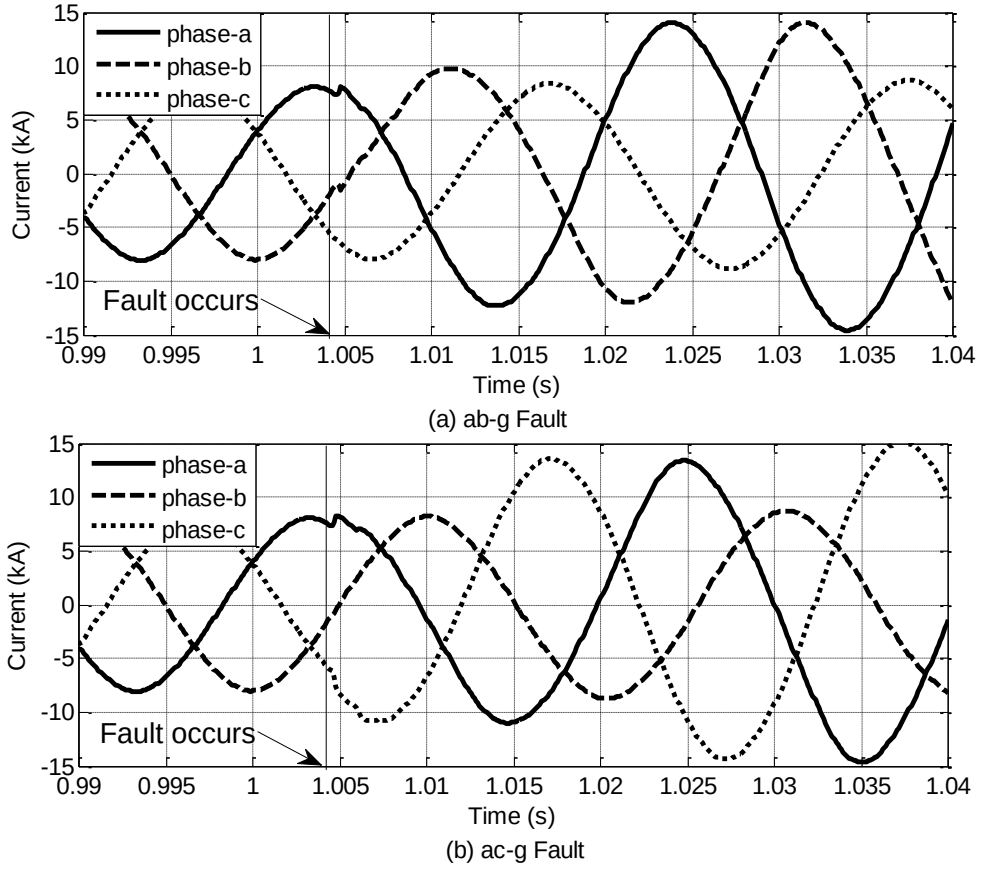
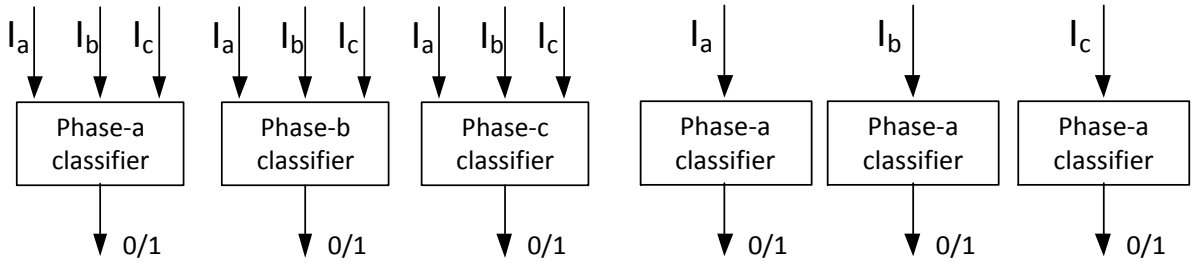


Fig. 5.5: Comparison of phase currents for (a) ab-g fault and (b) ac-g Fault.



(a) '3-ϕ' inputs. (b) '1-ϕ' inputs.

Fig. 5.6: Currents samples from (a) all the three phases given to each classifier, (b) individual phase given to corresponding classifier.

Table 5.2: Fault classification accuracy (%) for data window of 10 milliseconds

Training instances	Accuracy with RVM				Accuracy with SVM			
	1- ϕ 1 kHz	1- ϕ 4 kHz	3- ϕ 1 kHz	3- ϕ 4 kHz	1- ϕ 1 kHz	1- ϕ 4 kHz	3- ϕ 1 kHz	3- ϕ 4 kHz
500	93.53	94.69	97.04	97.22	93.38	97.45	98.46	98.61
1000	97.73	98.58	98.77	98.30	95.15	98.89	99.59	99.61
1500	98.51	98.17	99.12	99.28	95.18	98.49	99.64	99.64
2000	99.33	99.15	99.17	99.51	96.21	99.44	99.66	99.65
2500	99.65	99.40	99.63	99.65	97.71	99.12	99.74	99.89
3000	99.47	99.63	99.79	99.76	97.9	99.66	99.92	99.82
3500	98.57	99.69	99.76	99.81	98.2	99.69	99.85	99.90
4000	99.46	99.59	99.93	99.92	95.10	99.65	99.77	99.86
4400	99.61	99.83	99.95	99.91	97.67	99.60	99.87	99.93

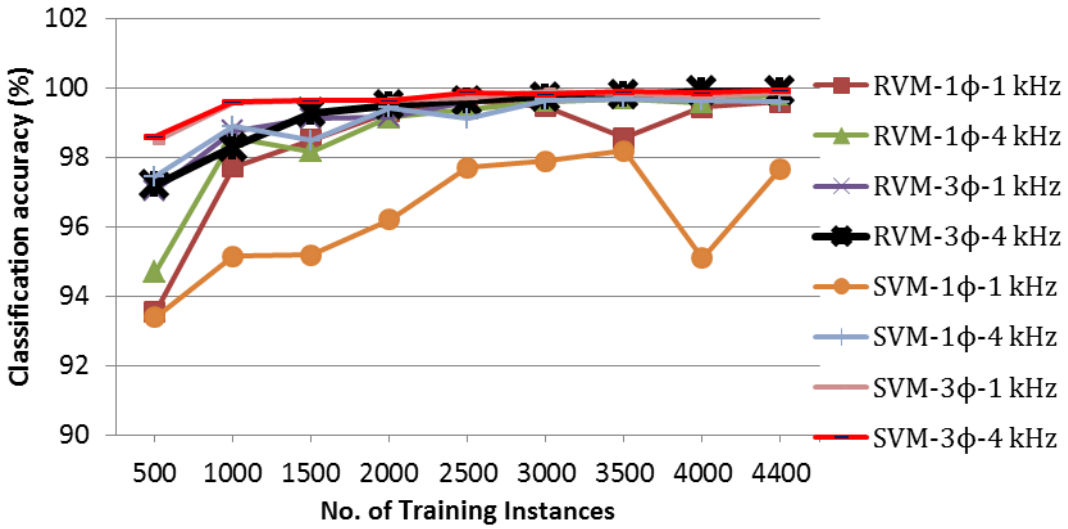


Fig. 5.7: Fault Classification Accuracy (%) for data window of 10 milliseconds.

5.6.3 Selection of Number of Instances for Training

To achieve higher testing accuracy, training set should be representative of the scenarios that will train model may face in the future (here during testing). Therefore, with an unbiased selection of training instances, greater the number of training instances, the training set is more likely to represent future scenarios. Consequently, more training instances (examples/scenarios/samples) are expected to give higher accuracy. However, higher numbers of training instances lead to higher training time. Therefore, to obtain higher accuracy with lower training efforts, accuracies of classifiers (SVM and RVM) were examined on 10000 testing instances after training classifiers at varying numbers of training instances (500, 1000, 1500, 2000, 2500, 3000, 3500, 4000, and 4400). Training and testing instances were unique and training set is disjoint from testing set. Nevertheless, testing set was kept same to offer a common point of reference for verifying accuracy of all models of classifiers (created by training with varying number of training instances).

With the increase in the number of training instances (remaining factors constant) fault classification accuracy increased up to 2000 instances and thereafter there is no or negligible improvement in accuracy (observed from Table 5.2 and Fig. 5.7). Therefore, 2000 instances seem optimal for obtaining higher accuracy with lesser training time.

5.6.4 Selection of sampling rate

For same time-length of the data window, data with higher sampling frequency is likely to have more information. If this information is relevant and discriminatory, then classification accuracy will be higher for the higher sampling rate. To verify this, fault accuracies at sampling rates of 1 kHz and 4 kHz were compared with each other. Signals sampled at 4 kHz have given higher accuracy most of the time, while all parameters except sampling rate are same. This can be observed by looking at Table 5.2, Fig. 5.7, and comparing contents of Table 5.3 and Table 5.4, Fig. 5.7 and Fig. 5.8, Fig. 5.9.

5.6.5 Selection of Data Window Length

Relevant information contained in the inputs to classifiers depends on the data window length of the signal. The information content of data window should increase with increase in data window length. However, classification accuracy is expected to be optimum for a particular window length beyond this length accuracy may decrease due to increase in irrelevant information.

As expected, fault classification accuracy improved with increase in data window length till data window length has not reached 10 ms (10 samples for 1 kHz and 40 samples for 4 kHz i.e. half cycle of system frequency). Therefore, the optimum length of data window can be asserted as half-cycle of power system frequency.

Table 5.3: Average Fault Classification Accuracy for varying window length at 1 kHz.

	Data window length (Samples per window)								
	4	**6**	**8**	**10**	**12**	**14**	**16**	**18**	**20**
RVM-1 ϕ	96.14	98.64	98.65	98.43	98.24	98.57	98.89	98.73	98.90
RVM-3 ϕ	97.46	99.23	99.17	99.24	99.13	99.33	99.18	99.31	99.46
SVM-1 ϕ	83.18	92.58	94.32	96.28	96.48	97.43	97.41	97.75	97.93
SVM-3 ϕ	97.40	99.23	99.53	99.61	99.61	99.67	99.69	99.64	99.63

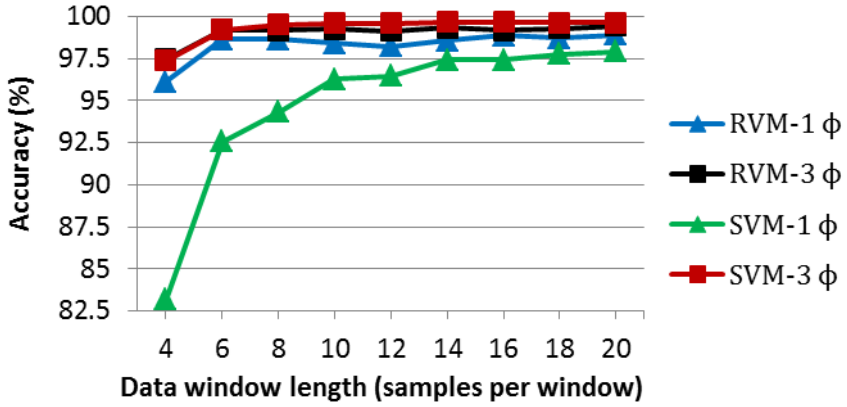


Fig. 5.8: Average Fault Classification Accuracy for varying window length at 1 kHz.

Table 5.4: Average Fault Classification Accuracy for varying window length at 4 kHz.

	Data window length (Samples per window)												
	20	**25**	**30**	**35**	**40**	**45**	**50**	**55**	**60**	**65**	**70**	**75**	**80**
RVM-1 ϕ	98.92	98.97	99.07	98.95	98.75	98.75	98.96	98.88	99.04	98.91	98.97	99.07	98.98
RVM-3 ϕ	99.23	99.16	99.27	98.97	99.26	99.33	99.07	99.28	99.22	99.22	99.20	99.24	99.25
SVM-1 ϕ	98.70	98.77	98.69	98.81	99.11	98.62	98.96	99.34	98.66	98.75	98.75	98.73	99.22
SVM-3 ϕ	99.49	99.63	99.65	99.58	99.66	99.70	99.63	99.67	99.67	99.72	99.68	99.66	99.63

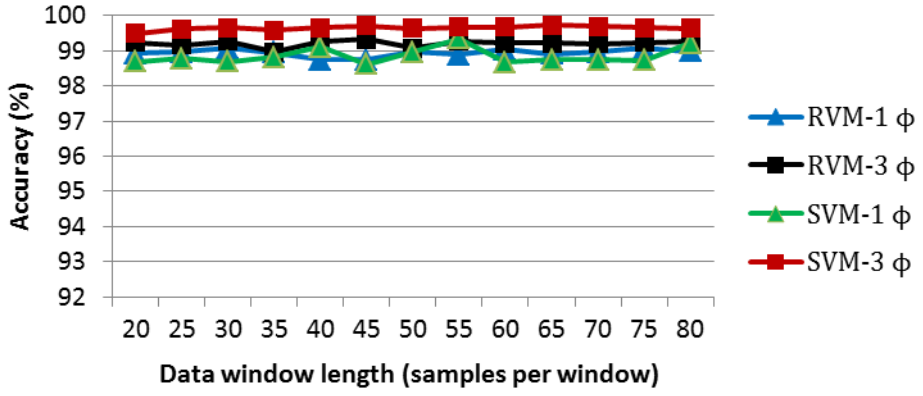


Fig. 5.9: Average Fault Classification Accuracy for varying window length at 4 kHz.

Table 5.5: Prediction time and number of support vectors (SVs) and relevance vectors (RVs) for the data window of 10 ms with 3-ϕ current at 4 kHz sampling frequency.

Training Instances	No. of RVs	No. of SVs	Time (ms per instance)	
			RVM	SVM
500	34	446	0.01322	0.23104
1000	36	621	0.01284	0.32371
1500	36	841	0.01261	0.44409
2000	38	1092	0.01177	0.57565
2500	36	1132	0.01155	0.59622
3000	39	1387	0.01133	0.74143
3500	41	1460	0.01205	0.77314
4000	38	1349	0.01138	0.71359
4500	40	1374	0.01189	0.72273
Average	37.56	1078	0.01207	0.56907

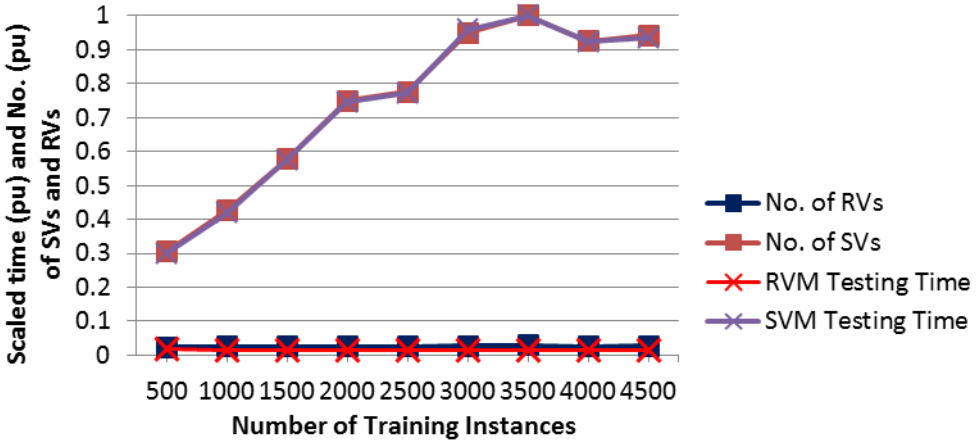


Fig. 5.10: Calculation time and number of support vectors (SVs) and relevance vectors (RVs).

5.6.6 Calculation Time for Fault-Type Identification

Although knowledge of fault type may be required for some offline applications, usually fault classification is an online application. Moreover, fault classification is a prerequisite for some protection algorithms. Therefore, the algorithm should be fast (few calculations) and data storage (memory) requirements should be small because of real-time nature of fault classification.

Calculation time of SVM and RVM for fault-type identification, depends on the number of hidden layer nodes i.e. support vectors (SVs) and relevance vectors (RVs) respectively. With the increase in the number of nodes (SVs or RVs) calculation time and data stored also increases (discussed in Section 3.7.2). To validate this postulate, number of nodes (SVs or

RVs) and testing time were recorded for classifier-models obtained by training on different number of training instances. Table 5.5 shows that calculation time increases with an increase in the number of SVs and RVs. Maximum number of nodes (1460 SVs) are in the SVM-model, trained with 3500 training instances, which also takes the maximum time (0.77314 ms) for the calculation. The relation between the number of nodes and calculation time can be easily visualized in Fig. 5.10 that plots scaled values (pu) of number of nodes (SVs / RVs) and calculation time against number of training instances. Here, numbers of nodes (SVs / RVs) are scaled by maximum number of nodes (1460 SVs) and calculation time is scaled by largest calculation time of classifiers (0.77314 ms).

Remarkably, smaller number of RVs and calculation time make RVM more suitable than SVM for real-time application of fault classification. On average RVM was 47 times faster and required 29 times lesser memory (nodes) than SVM (Table 5.5). Moreover, the number of RVs and calculation time of RVM-models remained almost constant with an increase in the number of training instances. In contrast, the number of SVs and calculation time of SVM-models increased with an increase in the number of training instances. Consequently, even if training set is large RVM can be successfully used to create a fault classifier with a smaller number of nodes suitable for real-time application. Nonetheless, training time of RVM is larger and increases with increase in size of the training set. However, large training time is not a limitation for RVM in this application, because training is performed offline. Out of this training comes an RVM-model that is faster and suitable for online implementation.

5.7 CONCLUSIONS

This chapter has used RVM and SVM for proposing new approaches for fault classification in series compensated transmission lines. RVM and SVM both achieved very high fault classification accuracy (more than 99.50%). Performance of RVM and SVM were compared on the basis of execution time and classification accuracy. Optimum values of sampling rate, data window length, and size of training set were deduced by analyzing large number (792) of experiments on simulated data. Half-cycle post-fault three phase current samples sampled at 4 kHz were found to give the best results. Suitable values of parameters of RVM and SVM were searched using cross-validation.

Fault classification accuracy of RVM is comparable to SVM. RVM required remarkably lesser data for storage and smaller calculation time than SVM. This made RVM more attractive than SVM for real time application of fault classification.

CHAPTER 6: FAULTY SECTION IDENTIFICATION

Identifying faulty section is a prerequisite for different protection algorithms. This chapter attempts to identify the faulty section with the help of Single-Hidden Layer Feedforward Networks (SLFNs). Performances of various SLFNs are evaluated on two different power systems, namely two-area system and 12-Bus system. Criteria for comparison of performance of SLFNs are section identification accuracy, training time, and prediction time. Depending on these criteria, suitability of SLFNs is discussed for its real-time or offline use.

6.1 INTRODUCTION

Identifying faulty section means evaluating whether the fault is in front of compensating bank or behind compensating bank. Due to issues discussed in Chapter 2, in series compensated lines identifying faulty section is challenging. Information of faulty section is required for different algorithms such as zone protection and fault location. Out of these algorithms, some are applied for real-time applications, for instance zone protection. For real-time applications section identification should be fast and storage requirements should be minimal, while there are other offline applications e.g. fault location, which can afford the luxury of slightly large calculation time and storage requirements [17]. However, accuracy is a prime concern for these algorithms. Nevertheless, faulty section identification is difficult in series compensated line because of the nonlinear nature of protective equipment of compensating bank [20, 28, 49, 95].

This chapter describes and discusses method for faulty section identification by using four popular Single-Hidden Layer Feedforward Networks (SLFNs) discussed in Chapter 1:. Building upon the discussions in previous chapters, this chapter compares the performances of these SLFNs on the basis of accuracy, training time, and testing time. Efficacies of these methods are demonstrated on the two-area system and the 12-Bus system. Appropriateness of every SLFN is deliberated for real-time and offline applications.

6.2 PROPOSED METHOD FOR FAULTY SECTION IDENTIFICATION

Learning from the experience of the previous chapter, this section proposes a method for identification of faulty section. Fig. 6.1 shows the overall procedure of faulty section identification. Initially current from all the three phases was measured through current transformer, then after passing this current through fourth order Butterworth filter current was sampled at 4 kHz. As pointed out in Section 5.6.2, current samples from all phases were preferred over samples from individual phases. Similarly, 4 kHz sampling rate was preferred over 1 kHz. Postfault half cycle current samples are given as input to the SLFN classifier. This choice of signal, sampling rate, and data window length for input is motivated from discussion of the previous chapter, because these choices are expected to have higher

content of relevant and discriminatory information, which can ultimately lead to better performance of SLFNs.

Because the concern is to ascertain whether the fault is in the section that is in front of or behind the compensating bank, classifier selected here is a two-class (binary) classifier. Zero at the output of the classifier indicates that the fault is in front of compensating bank (section 1), and one at the output denotes the fault behind the compensating bank (section 2).

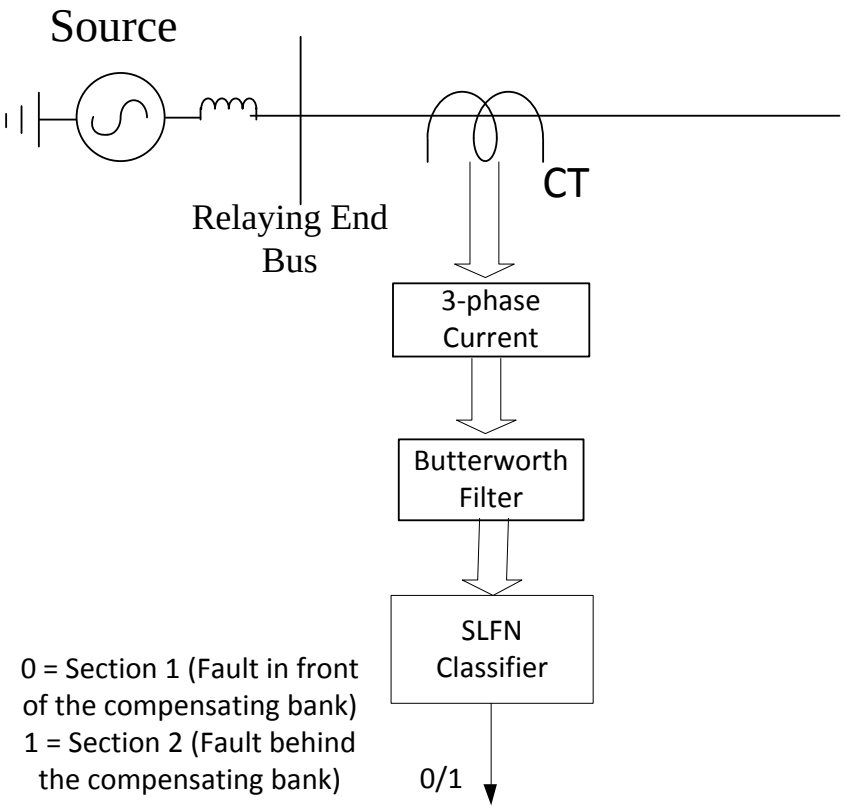


Fig. 6.1: Scheme for faulty section identification using SLFNs.

6.3 DISTRIBUTION OF DATA FOR CROSS-VALIDATION, TRAINING, AND TESTING

As described in Section 4.2, for two-area system, total 14,400 scenarios were simulated with different combinations (given in Table 4.1) of system and fault parameters. Out of these 14,400 scenarios half (7,200) were from the first transmission section (section in front of compensating bank) and half (7,200) were from the second section (section behind compensating bank). From these scenarios, 3000 from each section were taken for cross-validation and training, while remaining 4,200 scenarios from each section were taken for testing SLFNs. Thus, SLFNs were cross-validated and trained on total 6,000 instances and tested on total 8,400 instances. This distribution of data generated from simulation of two-area system is explained through the block diagram in Fig. 6.2.

Similarly, for 12-Bus system total 6,000 scenarios were simulated, 3,000 for each section (details in Section 4.3 and Table 4.2). From these simulated scenarios 4,000 (2,000

from each section) scenarios were taken for cross-validation and training, while remaining 2,000 (1,000 from each section) for testing the SLFN models. This distribution of fault instances of the 12-Bus system is illustrated in Fig. 6.3.

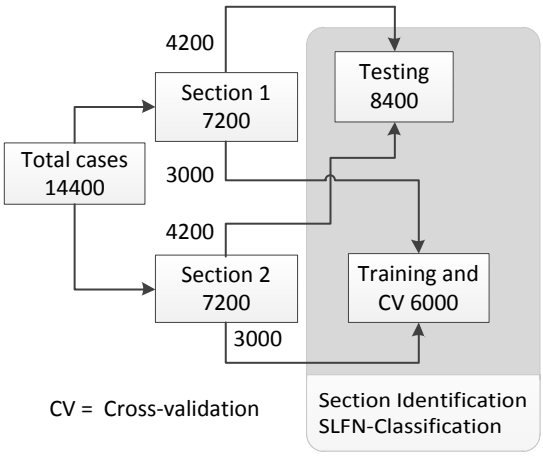


Fig. 6.2: Distribution of data for faulty section identification in two-area system.

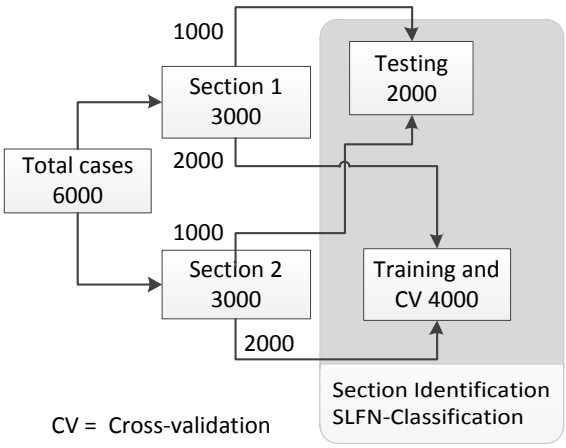


Fig. 6.3: Distribution of data for faulty section identification in 12-Bus system.

6.4 PARAMETER TUNING

As discussed in section 3.8, finding optimal value of parameters is critical for the performance of SLFNs. Therefore, five-fold cross-validation is used as a measure of performance of to search best value of parameters of SLFNs (except ELM). Here, all the SLFNs used radial basis function (RBF) as their activation function. RBF has one parameter, named basis width (σ). While RVM has no additional parameter to adjust, KELM and SVM have another parameter i.e. regularization parameter (C). Nonlinear grid search was used for searching best values of these parameters.

However, ELM does not require tuning of parameters, and selects all the parameters i.e. node vectors and basis width are selected randomly. Nevertheless, an ensemble of 100 ELM-models, was formed to average out variation in output (details in Section 3.8.3).

Apart from ELM, numbers of nodes of SLFNs are decided by their training procedure. In order to make ELM capable of mapping high level of nonlinearities, a high number of nodes are selected. Therefore, after a few trials, total 600 and 700 nodes were selected for two-area system and 12-Bus system, respectively.

6.5 RESULTS AND DISCUSSIONS

To demonstrate the efficacy of the proposed method, it was implemented on the two-area system and the 12-Bus system. This section gives details of the results obtained from experiments performed on data generated from simulation of these two systems. Results are further discussed to determine suitability of a classification algorithm (SLFN) for real-time and non-real-time applications.

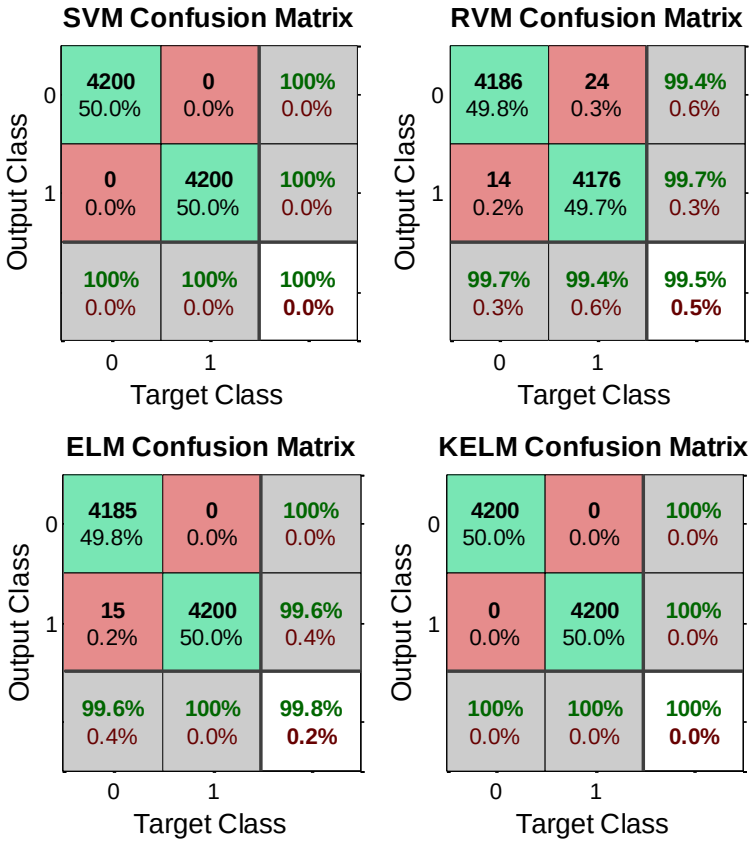


Fig. 6.4: Confusion matrix for two-area system.

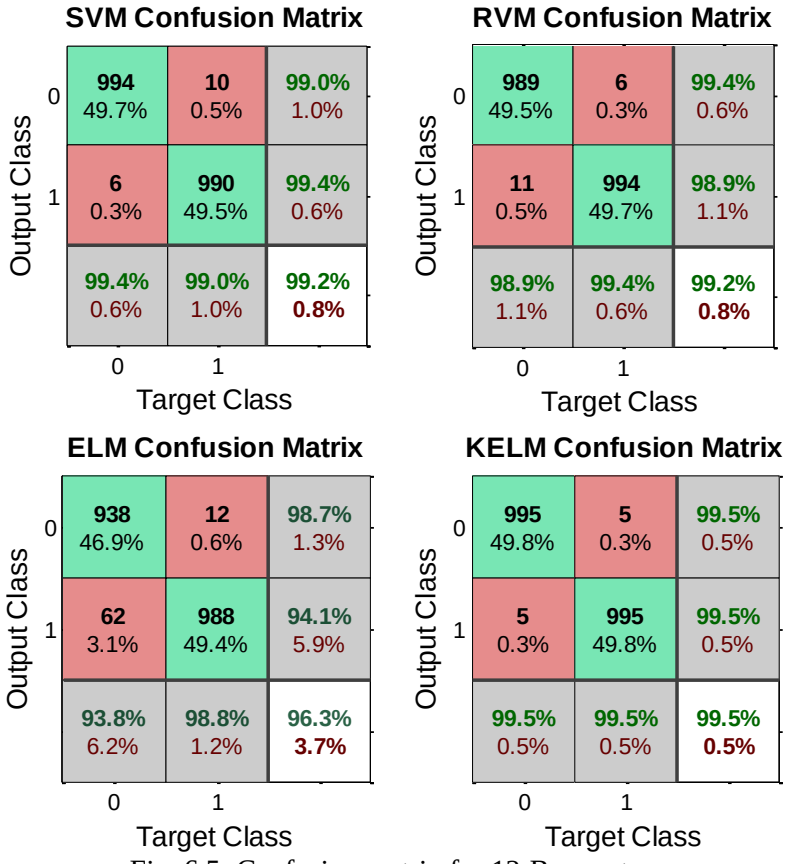


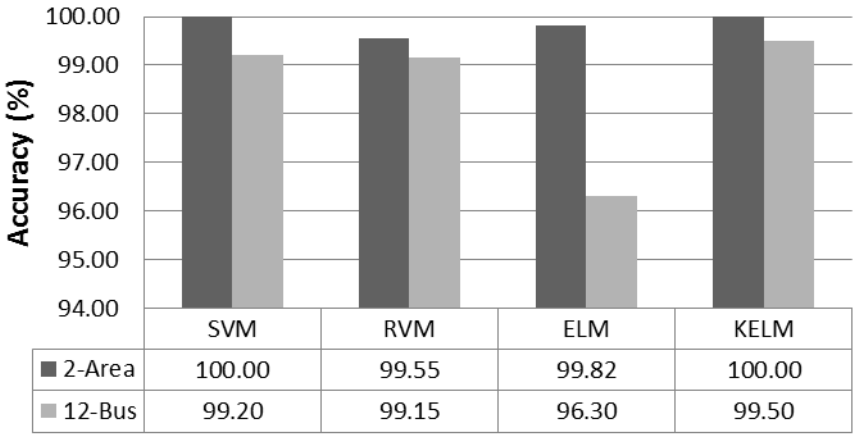
Fig. 6.5: Confusion matrix for 12-Bus system.

6.5.1 Accuracy of Faulty Section Identification

Fig. 6.4 and Fig. 6.5 show confusion matrices of two-area system and 12-Bus system respectively. Confusion matrices give the distribution of instances among real target classes and predicted classes of output from SLFN classifier, where actual target class is represented by columns and predicted class is represented by rows. Except bottom row and rightmost column, (i,j) element denotes number and percentage of instances that are predicted as i^{th} and their actual target was j^{th} class. Bottom-right block of confusion matrix gives overall percentage accuracy and error. Except bottom-right corner, lowest row gives percentage accuracy and error of a particular target class. Similarly, the extreme right column gives the percentage of correctly predicted outputs for a particular output class.

Confusion matrices of two-area system (Fig. 6.4) show that SVM and KELM have achieved perfect (100%) accuracies, while RVM and ELM have respectable accuracies. With 12-Bus system (Fig. 6.5), KELM has highest accuracy closely matched by SVM and RVM, while accuracy of ELM is comparatively small.

Moreover, Fig. 6.4 and Fig. 6.5 reveal that most of the errors of ELM are due to overreaching i.e. targets of '0' (fault in front of compensating bank) were predicted as '1' (fault behind the compensating bank). While other classifiers (specially KELM) seem unbiased from overreaching and underreaching.



	SVM	RVM	ELM	KELM
2-Area	100.00	99.55	99.82	100.00
12-Bus	99.20	99.15	96.30	99.50

Fig. 6.6: Faulty section identification accuracy (%).

Accuracies of faulty-section-classifiers (SLFNs) can be compared more compactly as shown in Fig. 6.6. This figure shows that, accuracies of all the SLFNs for the 12-Bus system are lesser than the two-area system. This decrease in accuracies can possibly be attributed to two reasons. Firstly, the 12-Bus system has detailed model and it is more complex than the two-area system. Secondly, the training instances of the 12-Bus system were lesser than the number of training instances of the two-area system. Therefore, probably smaller data contained smaller amount of relevant and discriminatory information.

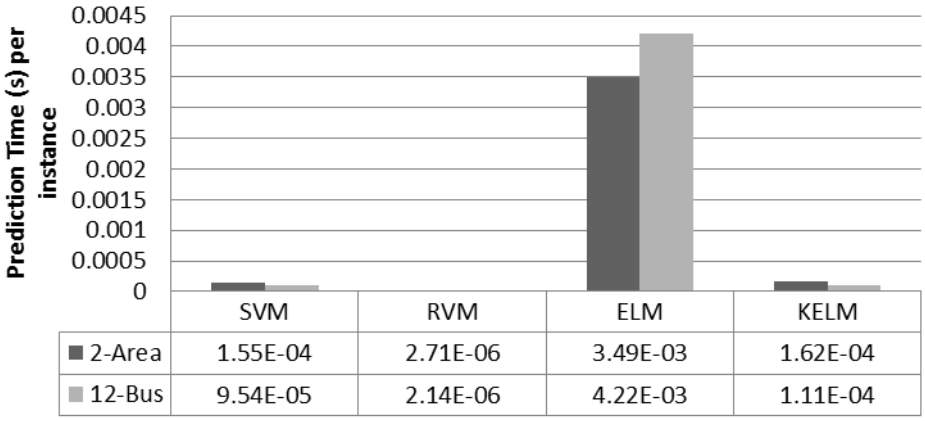
While accuracies of SVM, RVM, and KELM decreased slightly, accuracy of ELM degraded drastically (Fig. 6.6). In the wake of this, SLFNs other than ELM (particularly

KELM), can be stated as robust against the complexities of the system and can maintain their accuracy even with lesser amount of data.

6.5.2 Prediction Time and Number of Hidden Nodes

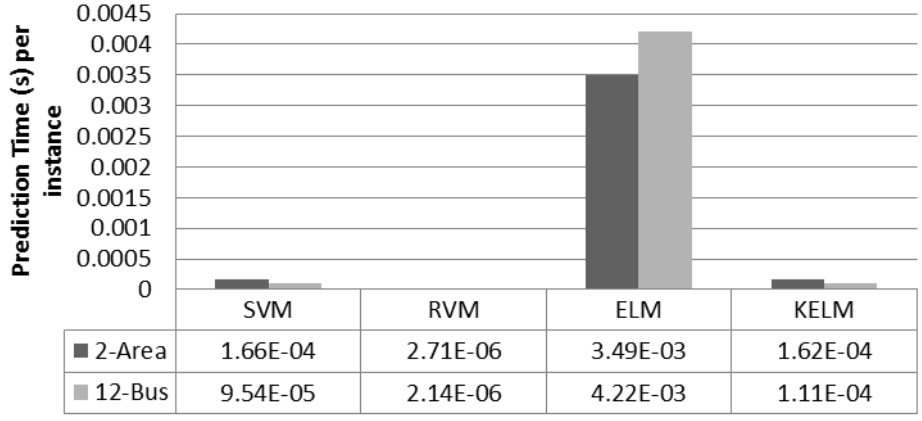
As pointed out in Section 3.7.2, prediction time and storage requirements of a SLFN model increase with the increase in the number of hidden layer nodes. Fig. 6.7 compares numbers of nodes of different SLFN models and corresponding time taken in prediction is given in Fig. 6.8.

As seen from these figures, RVM has the smallest number of nodes (relevance vectors), as a result prediction time of RVM is also smallest. Similar to RVM, noes of SVM are a subset of training data. However, SVM has more nodes (support vectors) than RVM (Fig. 6.7) and therefore prediction time of SVM is larger than RVM. KELM takes all training instances as its nodes. Therefore, KELMs has a larger number of nodes than SVM and RVM. Thus, prediction time of KELM is larger than SVM and RVM.



	SVM	RVM	ELM	KELM
2-Area	1.55E-04	2.71E-06	3.49E-03	1.62E-04
12-Bus	9.54E-05	2.14E-06	4.22E-03	1.11E-04

Fig. 6.7: Numbers of nodes of different SLFN models.



	SVM	RVM	ELM	KELM
2-Area	1.66E-04	2.71E-06	3.49E-03	1.62E-04
12-Bus	9.54E-05	2.14E-06	4.22E-03	1.11E-04

Fig. 6.8: Prediction time (seconds) per testing instance for different SLFNs.

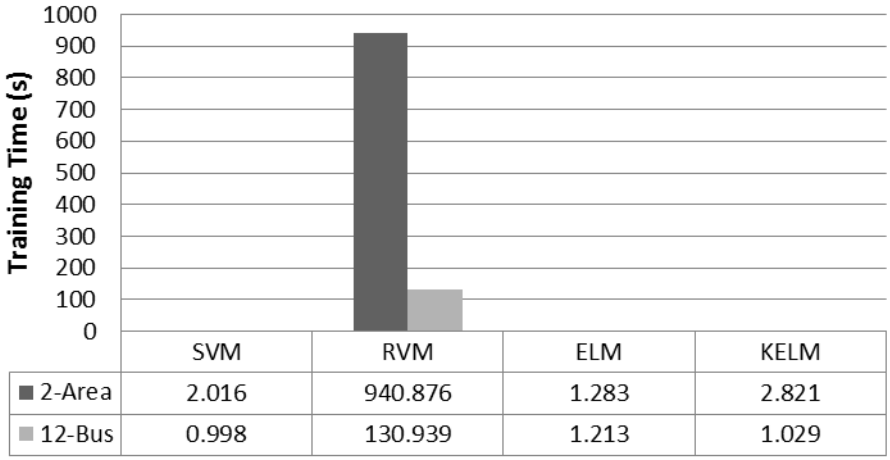
In our work single model (network) of ELM had nodes lesser than the total number of training instances (in practice ELM can have lesser or higher number of nodes than the number of training instances). However, due to the formation of an ensemble of 100 ELM-

models, total number of nodes increased 100 times i.e. 60,000 and 70,000 for two-area and 12-Bus system respectively. Consequently, prediction time of ELM was largest among all four SLFNs.

6.5.3 Time for Training and Parameter Searching

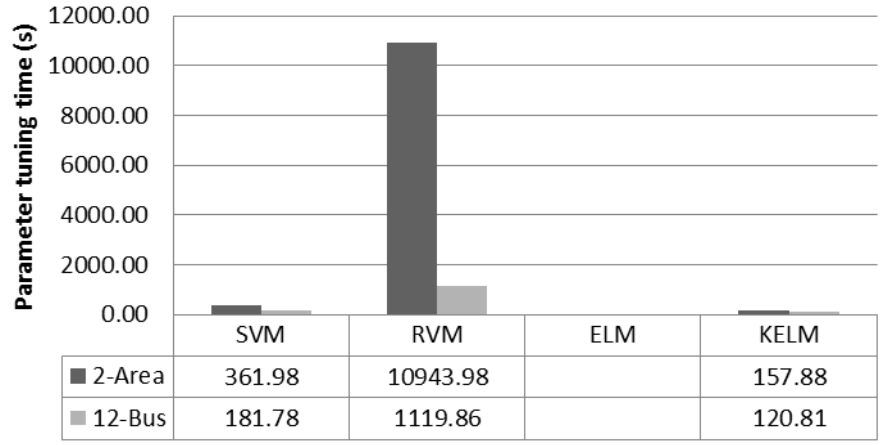
Training time of any SLFN classifier depends upon many factors such as the type of training algorithm, the number of training instances, complexity of the problem, and availability of relevant and discriminatory information. To check the performance of training algorithm, except training algorithm all other factors were kept constant.

From Fig. 6.9 it can be seen that the training time of RVM is significantly larger than training time of other SLFNs. Although, training time of single ELM is very small due to its training procedure (Section 3.4), due to the formation of an ensemble of 100 ELM-models training time becomes 100 times larger. Even in case of the 12-Bus system, training time of ELM exceeded training time of SVM and KELM. However, unlike other SLFNs, ELM does not require searching of values of parameters, which saves a lot of time. Therefore, overall time of ELM for model determination is small.



	SVM	RVM	ELM	KELM
2-Area	2.016	940.876	1.283	2.821
12-Bus	0.998	130.939	1.213	1.029

Fig. 6.9: Training time of SLFNs.



	SVM	RVM	ELM	KELM
2-Area	361.98	10943.98		157.88
12-Bus	181.78	1119.86		120.81

Fig. 6.10: Time taken for searching optimal value of parameters using cross-validation for SLFNs.

Time taken for parameter searching depends upon the number of folds of cross-validation, training time, and number of parameters to be searched. The smaller the amount of these factors smaller will be searching time. Although RVM has only one parameter to be searched, due to the extremely large training time of RVM, time for searching of parameter value of RVM is largest.

6.5.4 Suitability of SLFNs For Real-Time and Non-Real-Time Applications

This subsection gives general suggestions for appropriateness of SLFNs for real-time and non-real time applications based on the above-discussed results. However, suitability of SLFNs can heavily depend on specification of hardware on which they are to be implemented. For real-time applications, calculation time and storage requirement of the faulty section classifier should be as small as possible.

RVM is most suitable for real-time applications, as its prediction time (Fig. 6.8) and numbers of nodes (Fig. 6.7) are smallest among used SLFNs. Moreover, the accuracy of RVM is also high and comparable to SVM and KELM (Fig. 6.6). However, largest training (Fig. 6.9) and parameter tuning time (Fig. 6.10) of RVM makes RVM unsuitable for applications where training from updated data is required frequently.

SVM and KELM are suitable for offline applications due to their smaller training time (Fig. 6.9), parameter-tuning time (Fig. 6.10), and higher accuracy. However, prediction time (Fig. 6.8) and numbers of nodes (Fig. 6.7) of SVM and KELM are larger than RVM. Therefore, SVM and KELM should be used cautiously for online applications such as zone protection.

ELM has smaller training time (Fig. 6.9) and no parameter tuning is required, this makes ELM appropriate for applications where frequent training on updated data is required. However, due to formation of ensembles, ELM has relatively larger calculation time for prediction (Fig. 6.8), which can limit its use for real-time applications. Moreover, severely low accuracy (Fig. 6.6) of ELM for complex systems (similar to the 12-Bus system) can make it unattractive for both real-time and non-real time applications.

6.6 CONCLUSIONS

This chapter has presented a method for identification of the faulty section with the help of four popular SLFNs. Efficacies of these SLFNs were confirmed by implementing them on the two-area system and the 12-Bus system. The results of these implementations were discussed in detail and performances of four SLFNs were compared on the scale of accuracy, training time, storage requirements, and prediction time. From these results and comparisons, appropriateness of SLFNs was suggested for real-time and non-real-time applications. While RVM seems suitable for applications that require fast prediction with minimal storage requirements, KELM and SVM seem fit for applications with the requirement

of repeated training and highest accuracy. ELM is faster during training and has no parameter for searching. However, this could not make ELM useful either for real-time or non-real-time application. This was because of larger prediction time and poor accuracy of ELM on complex system.

Accurate fault location is vital in fast restoration of supply by helping fault-repairing crew to reach quickly at the location of the fault. This chapter presents a method for predictions of accurate location of the fault with the help of four popular SLFNs. The method is verified on two digitally simulated power systems. The chapter compares performances of these SLFNs with each other on the grounds of mean square error and time taken. Finally, by comparing performances, the chapter suggests suitable SLFN for fault location, which is a non-real-time application.

7.1 INTRODUCTION

After detection of fault in a transmission line, the protection system of transmission line separates the faulty section. This leads to disruption of supply to a large consumer base, consequently causing loss of revenue. For fast restoration of supply, knowledge of the exact location of fault is very helpful, so that a fault repairing crew can reach on the spot of fault to repair transmission line as early as possible. Manual inspection of transmission lines for locating fault is not pragmatic, as transmission lines may be hundreds of kilometers long. Therefore, efforts are made to locate fault from the fault signals recorded at the end of transmission lines. A fault location subroutine can come as an auxiliary subroutine to a relay or it can be a separate program executed on signals recorded from an advance digital fault recorder (DFR) [98, 99]. In both cases, fault location is a non-real-time application and executed after a fault has already occurred [100].

Fault location is inherently a difficult problem as the signal used for fault location depends upon many random variables such as fault inception time, fault resistance [101]. This problem is further aggravated in presence of series compensation, due to nonlinear operation of control and protective equipment of a series-compensating bank. Moreover, abrupt change in impedance seen at the point of the series compensating device and lack of knowledge of the faulty section makes fault location more challenging [37-39, 102].

This chapter explores four popular SLFNs (SVM, RVM, ELM, and KELM) for locating faults in series compensated line. Appropriateness of these SLFNs is checked for fault location (a non-real-time application) based on mean square error (MSE) and time taken for parameter tuning, training, and testing.

7.2 PROPOSED METHOD FOR FAULT LOCATION

Characteristics of fault signal changes abruptly for fault in front of compensating bank and fault behind the compensating bank [22, 28, 35, 95]. Therefore, constructing two SLFNs for fault location is expected to give better accuracy on account of availability of pertinent and cohesive information to individual SLFN. Inspired by this a two-step approach is adopted for fault location. In the first step, faulty section is identified from the method proposed in the previous chapter. In the second step, based on knowledge of faulty section, corresponding

SLFN is selected out of two SLFNs. SLFN in the first step is a classifier, while two SLFNs used in the second step are regression-SLFNs. This scheme is illustrated in Fig. 7.1.

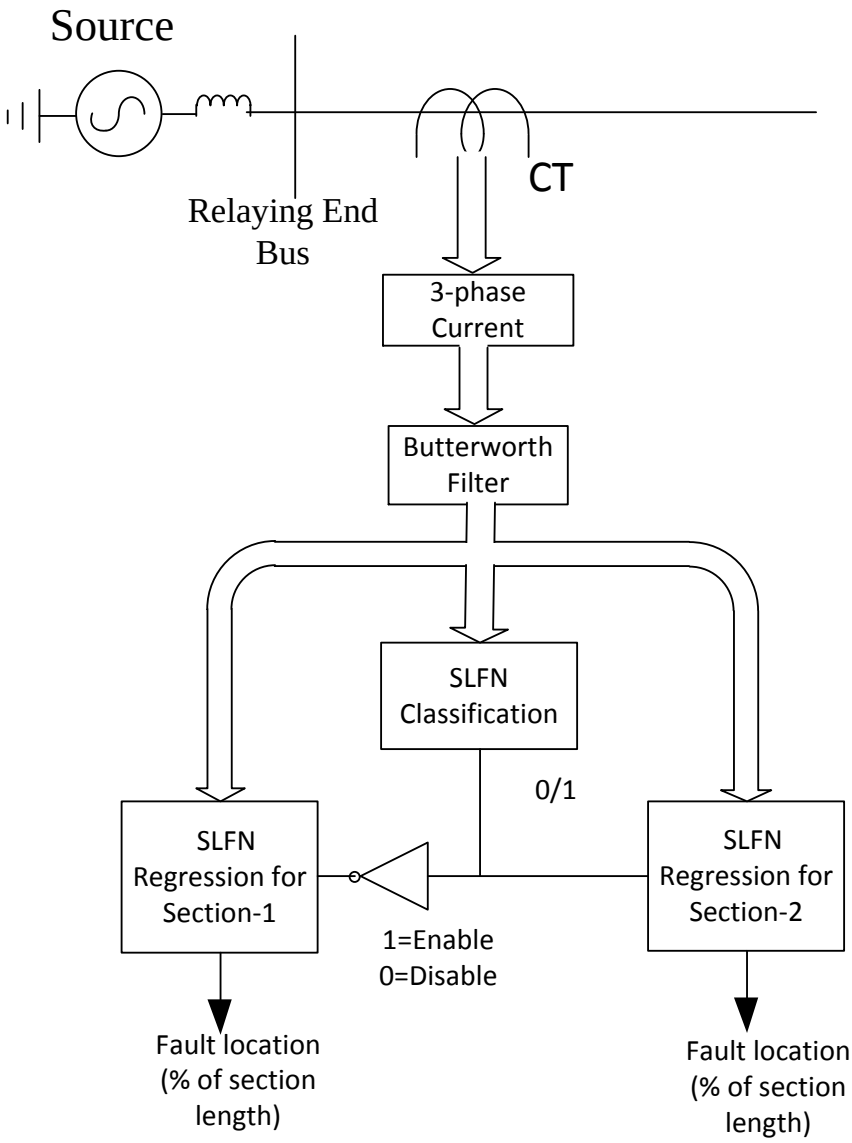


Fig. 7.1: Scheme for fault location in series compensated line.

After the occurrence of a fault, forty samples from half cycle of all the three phase current signals are given as input to SLFN-classifier for faulty section identification as well as to two-SLFNs for regression. If output of SLFN-classifier is '0' i.e. fault is in front of the compensating bank, then first SLFN for regression is enabled (second is disabled). While, '1' at the output of SLFN-classifier denotes that the fault is behind the compensating bank. Then, second SLNF for regression is enabled (first is disabled). Both the SLFNs for regression give output in the form of percentage of the length of the corresponding section.

To confirm the usefulness of the method, it was employed on two-area system and 12-Bus system. Before discussing the results, details of implementations such as distribution of data and parameter tuning are described in the following sections.

7.3 DISTRIBUTION OF DATA FOR CROSS-VALIDATION, TRAINING, AND TESTING

As stated earlier in Section 3.8, to give a correct assessment of the predictive performance of a learning algorithm, procedure of cross-validation, training, and testing are performed. This section describes allocation of all simulated scenarios (instances) for these procedures.

In case of two-area system, various combinations of fault and system parameters were taken to create 14,400 instances (details given in Section 4.2 and Table 4.1). Equal numbers of instances (7,200) were simulated for both sections. For cross-validation and training of section-identifier, 3,000 instances were selected randomly from each section (i.e. total 6,000) and testing was conducted on the rest of the 8,400 cases (half from each section). For SLFNs of fault location, 3,000 instances were chosen for cross-validation, and training from corresponding section, remaining instances (4,200) were taken for testing from the same section. This allocation of data is explained with the help of the block diagram in Fig. 7.2.

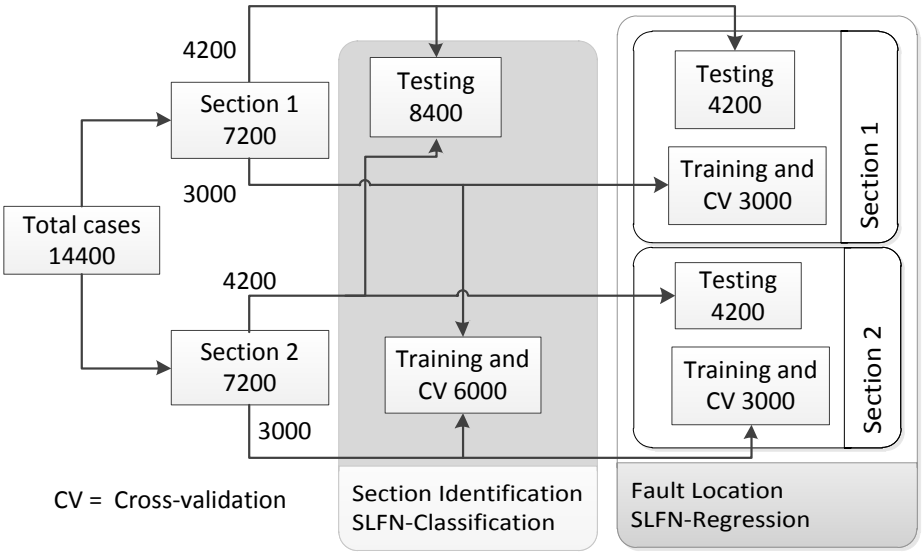


Fig. 7.2: Distribution of data for fault location in two-area system.

Similarly, allocation of data for the 12-Bus system is illustrated in Fig. 7.3. As described in Section 4.3 and Table 4.2, 6,000 instances were simulated using several combinations of parameters. Where, 3,000 instances were simulated for each section of transmission line. Total 4,000 instances were selected for cross-validation and training of section-identifier, (2,000 instances chosen randomly from each section) and remaining 2,000 instances were used for testing. For SLFNs of fault location, 2,000 instances were chosen for cross-validation, and training from corresponding section, remaining instances (1,000) of the same section were taken for testing.

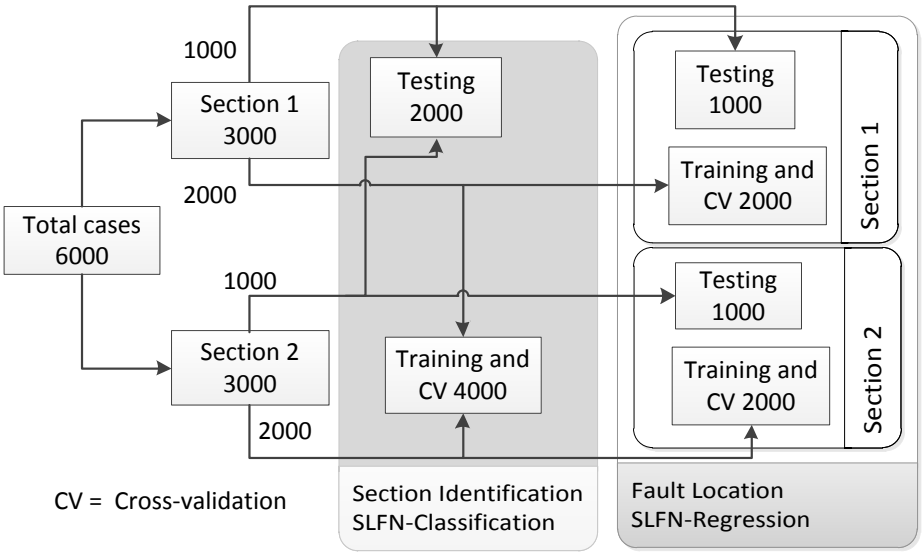


Fig. 7.3: Distribution of data for fault location in 12-Bus system.

Noticeably, training set of the first step (section identification) is union of training sets of the second step (fault location) and training sets of the second step are disjoint with each other. Similarly, testing set of the first step is union of the testing sets of the second step and testing sets of the second step do not have any common instance. Moreover, in every case training set is disjoint with testing set.

7.4 PARAMETER TUNING

For fault location, mean squared error (MSE) is considered as a measure of performance of SLFN during testing and cross-validation. For fault location MSE for *N* number of instances (cases) is defined as

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(\text{Per Unit Fault Location Error}_i)^2 \tag{7.1}$$

where Per Unit Fault Location Error for i^{th} instance is given by:

$$\text{Per Unit Fault Location Error}_i = \left| \frac{\text{Predicted Fault Locaion}_i \text{ - Actual (Target) Fault Location}_i}{\text{Total Length of Transmission Line}} \right| \tag{7.2}$$

Hereafter in this thesis, mean squared error during cross-validation (explained in section 3.8.2) will be termed as cross-validation-MSE or CV-MSE. For particular values of parameters, cross-validation-MSE gives a measure of the predictive performance of an SLFN for regression.

Searching set of parameter-values, which gives best (minimum) cross-validation-MSE is imperative for good performance of SLFN. For all SLFNs excluding ELM, a nonlinear grid search [77] with five-fold cross-validation was used to search optimal value of parameters.

Since ELM does not require parameter tuning, large numbers of nodes were taken to make ELM-models capable of mapping required nonlinearity. 1,500 and 2,000 nodes were taken for ELM-models of the two-area system and the 12-Bus system respectively. Moreover, an ensemble of 100 such ELM-model was constructed to avoid overfitting by averaging out the variation in output.

7.5 RESULTS AND DISCUSSIONS

To verify effectiveness of the SLFN-regressions, this section compares the performances of SLFNs with each other after applying them to the two-area system and 12-Bus system. Criteria for comparing performances are taken as MSE on output cases, cross-validation time, and training time. Based on these criteria the most appropriate SLFN is suggested.

7.5.1 Linear Curve Fitting of Predicted Locations versus Actual Location

Fig. 7.4 and Fig. 7.5 show scatter plot of predicted values of fault locations (Outputs Y) from SLFNs versus actual location (target T) for the two-area system and the 12-Bus system respectively. Results of fault location from both sections are combined in these figures. Moreover, outputs and targets are scaled by total length of transmission line of the corresponding system.

Besides, these figures show a red line of 'Y = T', preferably all plotted points should lie on this line ('Y = T'), because predicted locations (outputs) should be equal to actual locations (target). Further, a dashed line is plotted in these figures, this dashed line is obtained by linear regression of plotted points i.e. this line is obtained from linear curve fitting of the points. However, dashed line (obtained from curve fitting) is not visible as it is overlapped by ideal line ('Y = T'), because dashed line closely matches the ideal line. Value of regression coefficient (R) is also shown above the plots. R is the slope of the line obtained from linear curve fitting. The ideal value of R is one.

For the two-area system, Fig. 7.4 reveals that value of R for SVM and KELM is very close to one, most of the outputs are equal to their target value. However, in case of SVM predicted locations (outputs) near to relaying-end (near to T = 0) have larger error. Performance of RVM is not satisfactory as spread of points is comparatively more than other SLFNs. Although ELM seems to be performing well, error near to relaying end is very high.

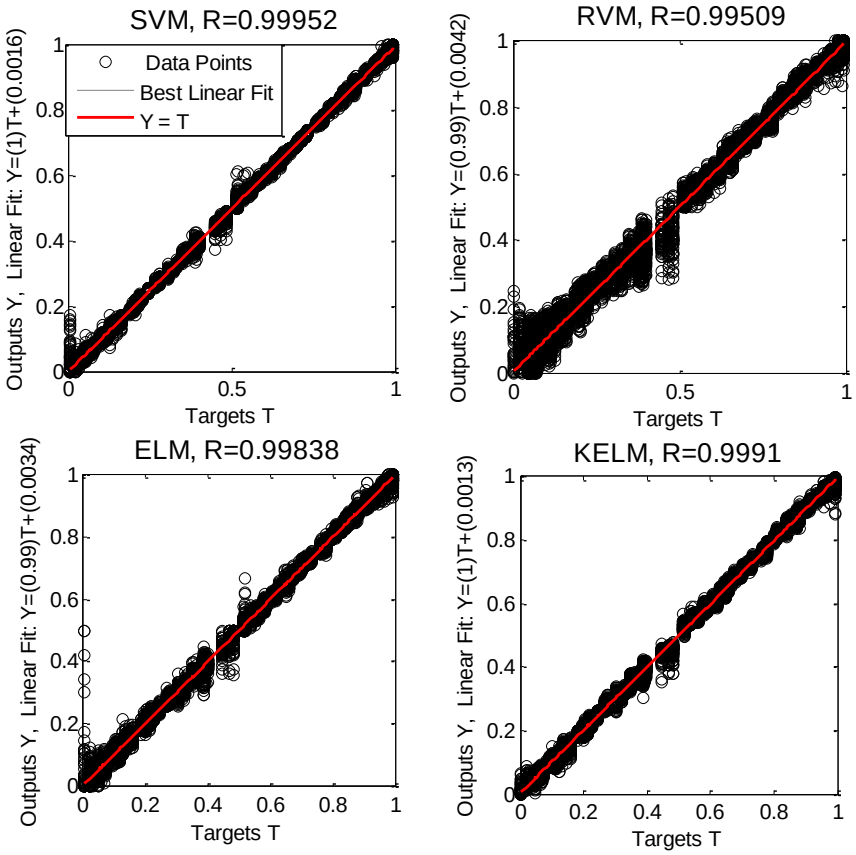


Fig. 7.4: Scatter plot of output fault location, against actual fault location for two-area system.

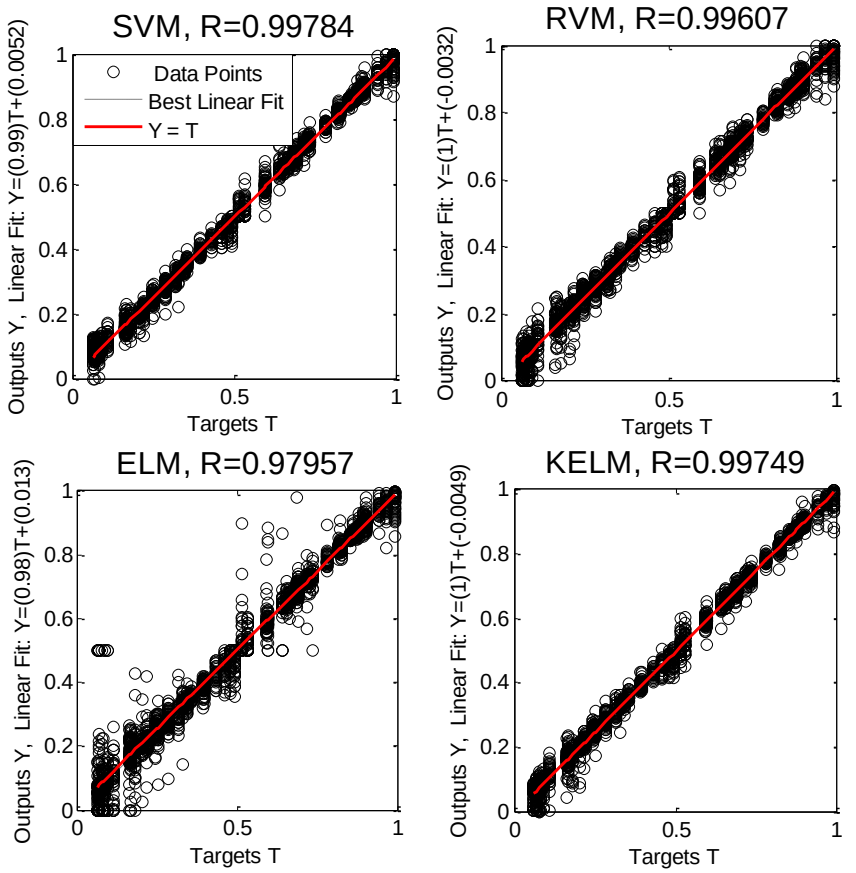
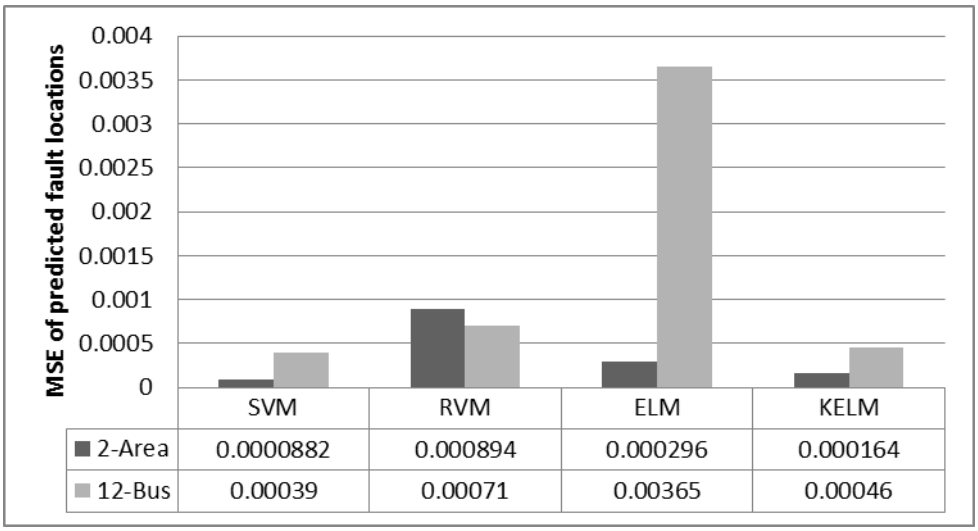


Fig. 7.5: Scatter plot of output fault location, against actual fault location for 12-Bus system.

For 12-Bus system, Fig. 7.5 shows that KELM and SVM have performed quite well. RVM has slightly better performance than its own performance in the two-area system. However, RVM has lesser value of R than SVM and KELM. Nevertheless, ELM has very poor performance. For ELM with the 12-Bus system, the error is so high that for some points, fault locations near to one end of transmission line section are predicted as fault at the opposite end of the same section.

7.5.2 Mean Squared Error (MSE) of Predicted Fault Locations

Mean squared errors (MSEs) of predicted fault locations are presented in Fig. 7.6 with the help of bar chart, the corresponding values are embedded in same figure using table. MSE is calculated with the help of the formula given in (7.1). From the figure, it can be seen that SVM has given minimum MSE and the performance of SVM is closely matched by KELM for both of the systems (two-area and 12-Bus). RVM was not able to match performances of KELM and SVM. Similar to the faulty section identification, for the 12-Bus system, performance of ELM has degraded drastically i.e. MSE is very high. Same inference can be drawn for poor performance of ELM, viz, ELM does not seem capable of mapping complexities of detailed 12-Bus system with lesser number of training instances.



	SVM	RVM	ELM	KELM
2-Area	0.0000882	0.000894	0.000296	0.000164
12-Bus	0.00039	0.00071	0.00365	0.00046

Fig. 7.6: Mean squared error (MSE) of predicted fault locations.

7.5.3 Maximum Fault Location Error

Fig. 7.7 compares maximum fault location errors (given by (7.2)) of SLFNs for the two systems under study. This figure reveals that maximum fault location error is least for KELM and largest for ELM. From Fig. 7.7 it can be observed that for ELM maximum error is more than 95% of one section. Further, SVM and RVM have maximum error higher than KELM's maximum error and lower than KELM.

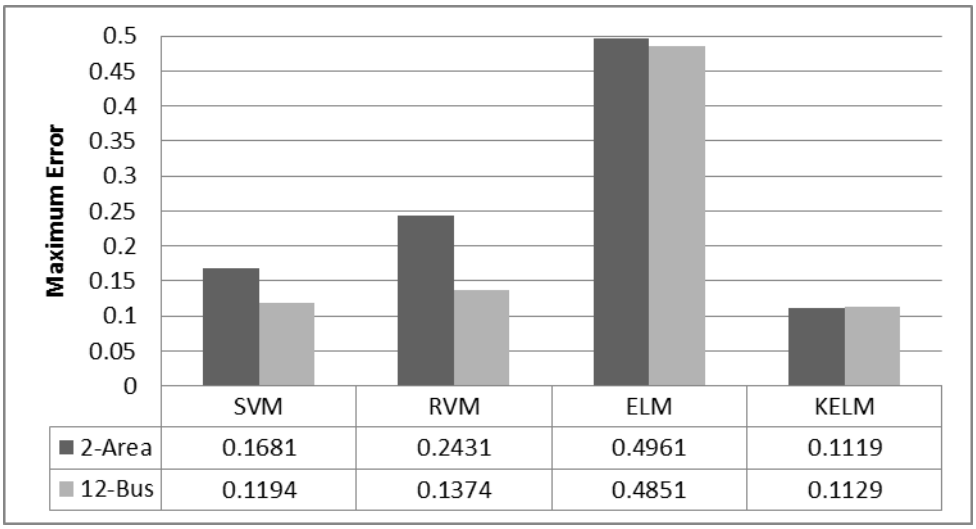


Fig. 7.7: Maximum fault location error.

7.5.4 Time for Cross-Validation, Training, and Prediction

This section compares performances of all the SLFNs on the basis of time taken by these SLFNs for cross-validation, training, and prediction. Although values of time taken are dependent on hardware platform on which they are implemented, comparative performance SLFNs will remain similar as their algorithms will remain the same. However, for reference specification of machine on which experiments were performed are given in Appendix-A3.

Since fault location is a non-real-time application, calculation time and storage requirements are not as stringent as they are for real-time applications like zone protection.

If trained model is already available then prediction time is not significant for non-real time applications. Fig. 7.8 shows that even maximum prediction time (taken by ELM) is in order of milliseconds. Large prediction time of ELM is due to formation of ensembles (details given in Section 3.8.3).

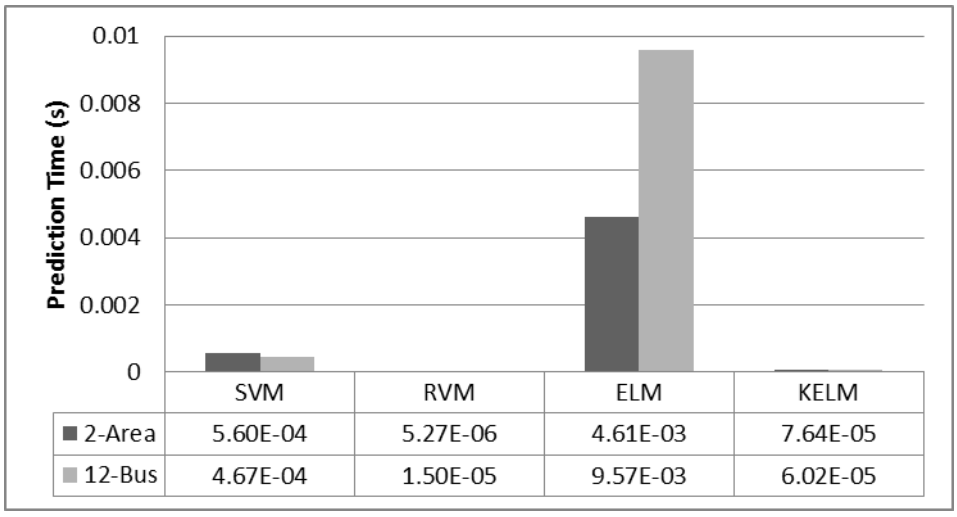


Fig. 7.8: Prediction time (s) of SLFNs for fault location.

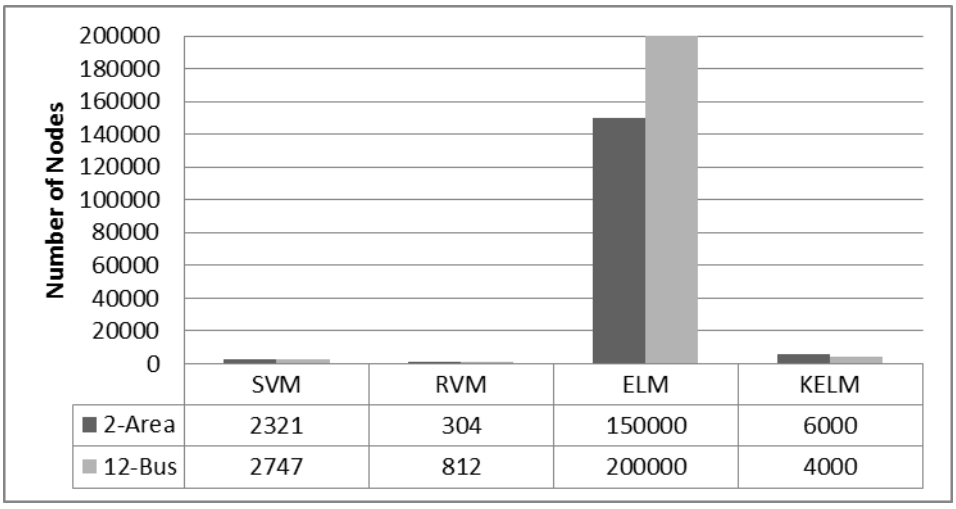


Fig. 7.9: Number of hidden layer nodes of SLFN-models trained for fault location.

Similarly, for already trained model storage requirements are not a limitation. Fig. 7.9 gives number of nodes of SLFNs, as described earlier in section 3.7.2, higher number of nodes leads to higher prediction time. Therefore, pattern of number of nodes plotted in Fig. 7.9 is matching with pattern of prediction time plotted in Fig. 7.8.

If training of SLFN is required frequently due to recurrent update of training data then time for parameter searching and training starts playing a significant role. Fig. 7.10 compares training time of different SLFNs for the two systems. This figure shows that RVM has taken maximum time for training (lesser than 3 minutes). However, even this training time can be stated as affordable for fault location.

Fig. 7.11 provides time taken for searching optimal values of parameters. Nonlinear grid search is performed to find optimal values of parameters for all SLFNs (apart from ELM). While ELM does not require parameter searching, large errors for fault locations by ELM render ELM unusable for fault location.

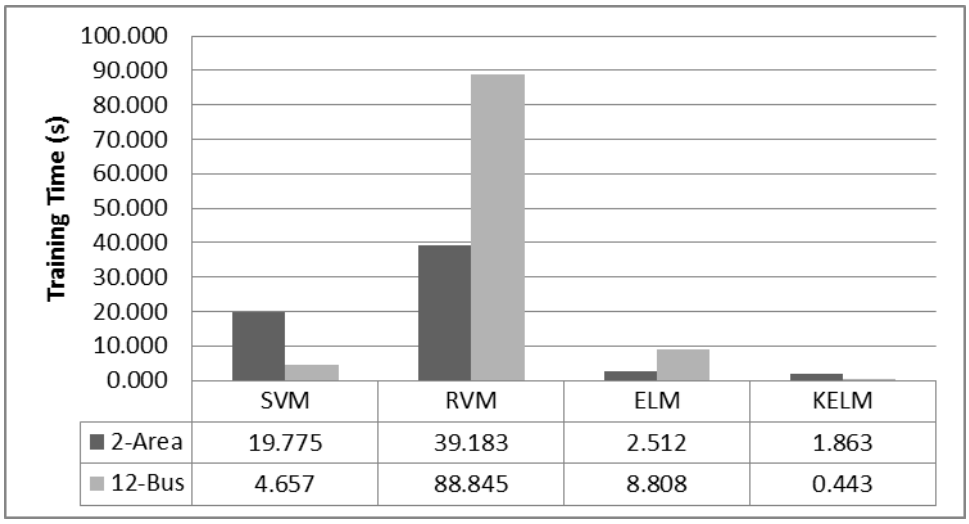


Fig. 7.10: Training time of different SLFNs for fault location.

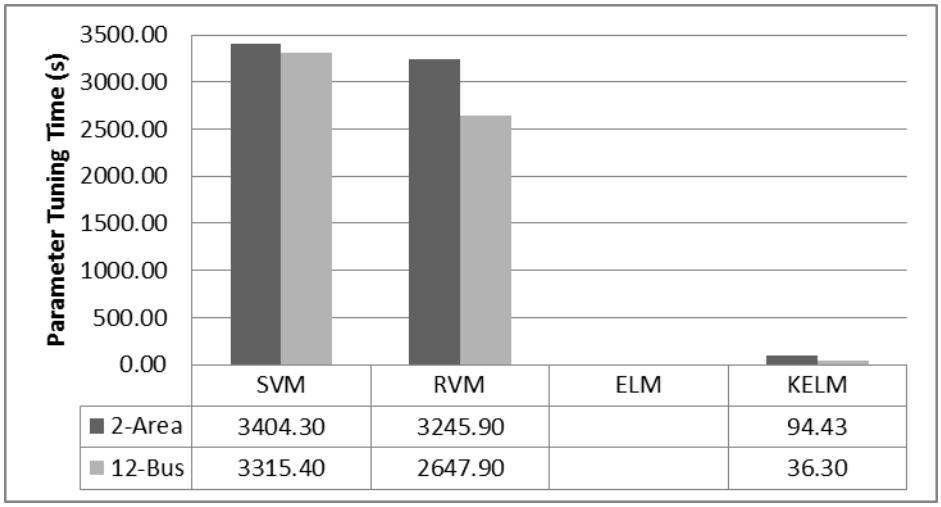


Fig. 7.11: Time taken by various SLFNs for tuning their parameters.

Although RVM has only one parameter to tune, it takes quite high time (nearly one hour) for tuning only one parameter. Unlike the case of section identification, parameter-tuning time of SVM is very high (nearly one hour), even more than RVM (observed from Fig. 7.11). This increase in parameter-tuning time may be attributed to complexity of fault location problem and additional parameter (ϵ), as additional parameter has increased combinations of parameter-values exponentially due to nonlinear grid search.

Noticeably, KELM has very small training and parameter-tuning time (Fig. 7.11). Moreover, KELM has smallest maximum-fault-location-error amongst used SLFNs (Fig. 7.7). In addition, MSE of KELM is very small and competitive to SVM (Fig. 7.6). Due to these reasons, KELM seems most suitable SLFN for fault location purpose.

7.6 CONCLUSIONS

This chapter proposed a two-step approach for locating fault in series compensated line. First step finds faulty section of the line using method described in previous chapter. Second step finds precise location of fault within the faulty section identified in the first step. Four popular SLFNs were used for regression to find location of fault. Efficacy of the proposed approach with these SLFNs was ascertained by implementing it on the two-area system and the 12-Bus system.

Although overall time taken by ELM was small, MSE of ELM was very poor. MSE of RVM and time taken by RVM were not satisfactory. Although SVM achieved minimum MSE, its parameter-tuning time was largest. This left SVM inconvenient if repeated training is required. KELM has closely matched MSE of SVM. Moreover, overall time taken by KELM for parameter-tuning, training, and testing was very small compared to RVM and SVM. This made KELM most appropriate for application of fault location, which is a non-real-time application.

CHAPTER 8: FAULT LOCATION WITH MODIFIED SEQUENCE OF INPUT

As discussed in previous chapter, accurate fault location is essential for reduction of time and effort of fault-repairing crew to restore power supply. This chapter presents a new approach for estimating accurate fault location in series compensated line. In this method, based on the type of the fault, four SLFNs for each section are used to predict location of fault. Moreover, order of inputs to the fault predictor is modified according to their type of fault. This resulted in improved accuracy of fault location.

8.1 INTRODUCTION

Usually series compensation is provided on transmission lines carrying large power. Therefore, disruption of power supply due to disconnection of series compensated line can result in huge loss of revenue. Information of accurate fault location reduces time of disruption of supply, thereby reduces loss of revenue. Therefore, in previous chapter a new method for fault location has been proposed, which used information of faulty section to improve accuracy of KELM for fault location. A large data set was generated for this study, and out of this data set, instances for training and testing set were selected randomly. Although, every instance of the generated data set was unique, ranges of parameter for generating data was same. Good accuracy was achieved using this method. However, accuracy of the method discussed in previous chapter decreased slightly when its performance was checked on different data set with different range of parameters.

To achieve further improved accuracy, in this chapter, a new approach is proposed that utilizes knowledge of fault-type along with faulty section. Besides sequences of input to the KELM-fault-locator have been modified according to type of fault (L-G, LL-G, LL, LLL/LLL-G). Again, KELM is compared with other SLFNs namely, SVM, RVM, and ELM. Effectiveness of each SLFN is measured based on mean square error (MSE), maximum error, median error, and time taken for parameter tuning, training, and testing.

8.2 PROPOSED METHOD FOR FAULT LOCATION

Features of fault signals are dependent on the type of fault (L-G, LL-G, LL, LLL/LLL-G) and on the section at which the fault occurs [22, 28, 35, 95]. Therefore, if type and section of fault are already known, then training different SLFN for every fault-type of both sections (i.e. total 8 SLFNs) is expected to give better accuracy. This is due to availability of relevant and consistent information to individual SLFN. Motivated by this, a new approach for fault location is proposed, which uses eight SLFNs as shown in Fig. 8.1.

Initially for anti-aliasing purpose, current signals from CTs were passed through a third-order Butterworth filter, then currents signal were sampled. Forty samples of current from post-fault half cycle were taken from all the three phases i.e. dimension of input vector ($I_{3\phi}$) to SLFNs is 120. Therefore, sampling frequencies for 50 Hz (Case I, Section 4.2) and 60

Hz (Case II, Section 4.3) systems are taken as 4 kHz and 4.8 kHz respectively. In compliance with the Nyquist–Shannon sampling theorem, cutoff-frequencies of anti-aliasing filters were half of the sampling rate of the signal i.e. cutoff-frequencies for 50 Hz and 60 Hz power system are 2 kHz and 2.4 kHz respectively.

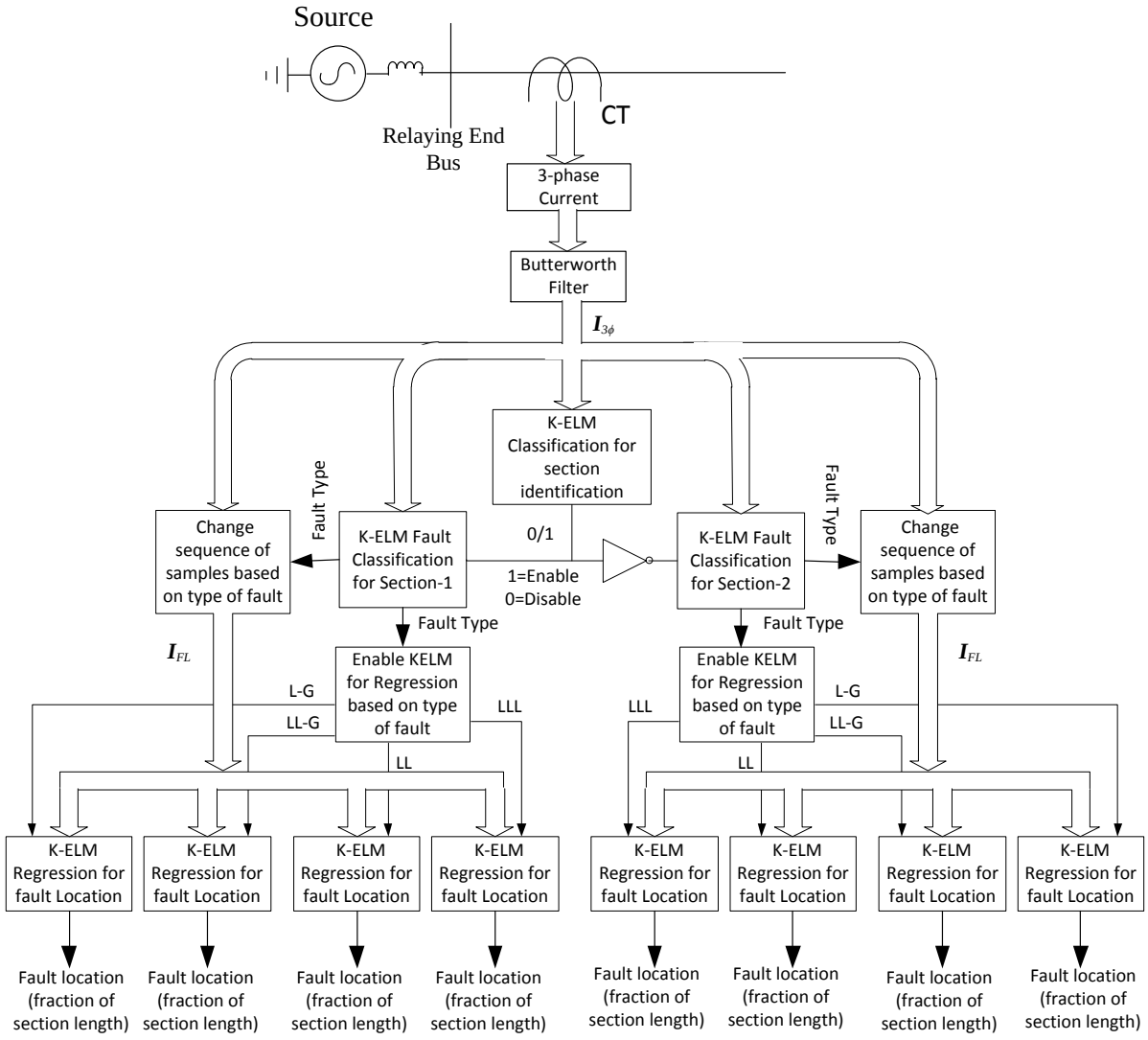


Fig. 8.1 Proposed scheme for fault location in series compensated line.

After this, faulted section has been identified by using KELM classification, for this purpose, vector of current samples ($\boldsymbol{I}_{3\phi}$) is given as input to the KELM-classifier that gives the output in binary form. Where one and zero indicate, fault has occurred before and after the TCSC respectively. If the output of KELM-classifier is one it enables further steps of the algorithm for the first section, otherwise it enables further steps of the algorithm for the second section.

In the next step, enabled KELM for fault classification of faulty section predicts type of fault by taking same vector of current samples ($\boldsymbol{I}_{3\phi}$). Method used for fault detection is schematically similar to Fig. 5.3, except that instead of zero sequence current, residual

current is used for ground classification. Three classifiers belonging to each phase are directly given three phase current samples, while for ground classifier residual current samples are taken as input. Input vector of residual current is defined as:

$$\boldsymbol{I}_{Res} = \boldsymbol{I}_a + \boldsymbol{I}_b + \boldsymbol{I}_c$$

where $\boldsymbol{I}_a$, $\boldsymbol{I}_b$, and $\boldsymbol{I}_c$ are 40 dimensional column vector of current samples from phase-a, phase-b, and phase-c respectively. Output of these classifier is binary, where 0 indicates and 1 denotes no fault in that phase or ground. Then, based on the combination of these outputs, type of fault is decided as shown in Table 5.1.

Now, based on type of fault (L-G, LL-G, LL, and LLL) one KELM-Regression is enabled for fault location. Input vector ($\boldsymbol{I}_{FL}$) to this enabled KELM-Regression is obtained by modifying the order of the samples of current on the basis of type of fault as shown in Table 8.1. Modification of fault sample order makes individual fault type effectively similar to other fault type within subtype (L-G, LL-G, LL) of fault. For example, by changing the input vector for c-g type of fault according to Table 8.1, modified vector becomes equivalent to input vector of a-g type of fault for 'L-G fault locator'. During the experiments, this was found to improve the accuracy for fault location.

Finally, enabled KELM-regression block predicts fault location as a fraction of the length of faulted section.

Table 8.1 Formation of input vector to KELM for fault location based on type of fault.

Type of faults	a-g	b-g	c-g	ab	bc	ca	ab-g	bc-g	ca-g	abc/ abc-g
Vector of current samples ($\boldsymbol{I}_{FL}$)	$\begin{bmatrix} I_a \\ I_b \\ I_c \end{bmatrix}$	$\begin{bmatrix} I_b \\ I_c \\ I_a \end{bmatrix}$	$\begin{bmatrix} I_c \\ I_a \\ I_b \end{bmatrix}$	$\begin{bmatrix} I_a \\ I_b \\ I_c \end{bmatrix}$	$\begin{bmatrix} I_b \\ I_c \\ I_a \end{bmatrix}$	$\begin{bmatrix} I_c \\ I_a \\ I_b \end{bmatrix}$	$\begin{bmatrix} I_a \\ I_b \\ I_c \end{bmatrix}$	$\begin{bmatrix} I_b \\ I_c \\ I_a \end{bmatrix}$	$\begin{bmatrix} I_c \\ I_a \\ I_b \end{bmatrix}$	$\begin{bmatrix} I_a \\ I_b \\ I_c \end{bmatrix}$
Input to KELM for	L-G faults.			LL-G faults.			LL faults.			LLL/ LLL-G faults.
	where I_a, I_b, and I_c are 40 dimensional column vectors consisting of post fault current samples from phase-a, phase-b, and phase-c respectively.									

To validate the practicality of the method, it was employed on a two-area system and 12-Bus system. Details of implementations such as distribution of data and parameter tuning are described in the next sections, before discussing the results.

8.3 GENERATION AND DISTRIBUTION OF DATA FOR CROSS-VALIDATION, TRAINING, AND TESTING

As stated earlier in Section 3.8, to give a correct assessment of the predictive performance of a learning algorithm, procedure of cross-validation, training, and testing are performed. This section describes parameter variation for the simulation of different scenarios (instances) for these procedures. It is noticeable that same data set is used for cross-validation and training, while separate data set is generated for testing.

In case of two-area system described in Section 4.2, five thousand fault scenarios were simulated for training and cross-validation with the variation in parameters as shown in Table 8.2. Same table shows, combination of parameters for separately simulated ten thousand fault scenarios.

Table 8.2 Variation of parameter-values for two-area system.

SNo.	Parameter	Variation Type	Values/Rage of Parameters for Training Set	No. of Variations for Training Set	Values/Rage of Parameters for Testing Set	No. of Variations for Testing Set
1	Sections	Sequential	Two sides of TCSC	2	Two sides of TCSC	2
2	Fault Type (10 Types).	Sequential	A-G, B-G, C-G, AB, BC, CA, AB-G, BC-G, CA-G, ABC/ABC-G	10	A-G, B-G, C-G, AB, BC, CA, AB-G, BC-G, CA-G, ABC/ABC-G	10
3	Firing angle (Compensation %)	Random	**140° to 180° (30% to 40%)**	250	136° to 180° (30% to 60%)	500
4	Fault resistance	Random	**0 Ω to 50 Ω**		0 Ω to 100 Ω	
5	Fault inception angle:	Random	0° to 360°		0° to 360°	
6	Load angle of source at bus 1	Random	**15° to 25°**		10° to 30°	
7	Source impedance	Random	85% to 115% (for both sources)		70% to 130% (for both sources)	
8	Fault Locations	Random	0 to 150 km on both sides of TCSC		0 to 150 km on both sides of TCSC	
	Total cases			**2x10x250 = 5000**		**2x10x500 = 10000**

Except for first two parameters in the table, values of other parameters in table were selected randomly (uniform random distribution) within their specified range. This was to ensure that each instance is different from other. Further, different seed values are taken to ensure that random values produced in the testing set are different from the training set. Moreover, it was verified that parameter set for each instance is unique.

In this way, on both sides of TCSC for every ten types of fault, 250 cases were simulated for training and 500 cases for testing purpose i.e. total 5000 cases were simulated for training and 10000 cases for testing purpose. Distribution of 5,000 training cases and 10,000 testing cases is shown in Fig. 8.2 and Fig. 8.3 respectively.

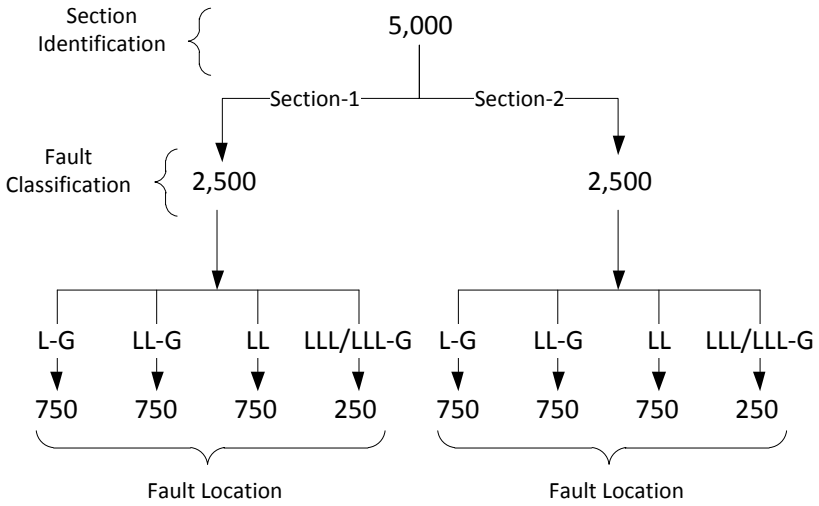


Fig. 8.2 Distribution of 5,000 simulated cases for training and cross-validation of SLFNs.

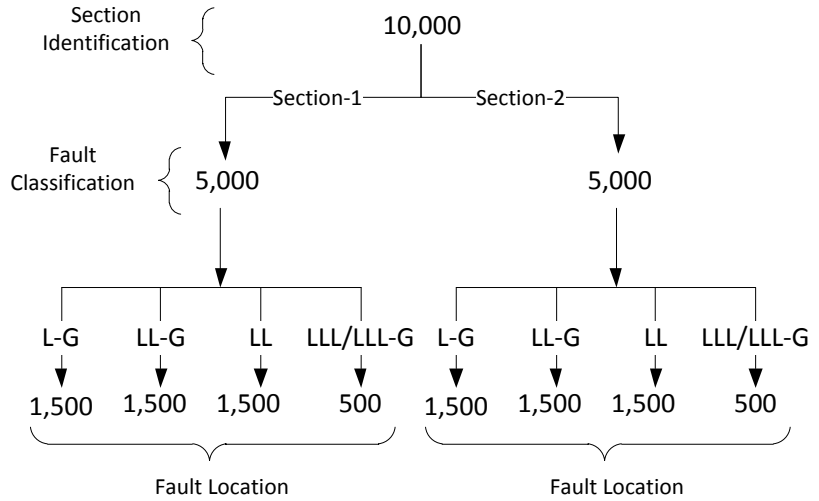


Fig. 8.3 Distribution of of 10,000 simulated cases for Testing SLFNs.

Likewise, for the **12-Bus System** (Section 4.3) total 5,000 cases were simulated for training (cross-validation) and 10,000 cases for testing purpose. Variation of parameters in this case is given in Table 8.3. These simulated instances are allocated for training (Fig. 8.2) and testing (Fig. 8.3) similar to the two-area system.

Table 8.3 Variation of parameter-values for 12-Bus system.

S.No.	Parameter	Variation Type	Values/Rage of Parameters for Training Set	No. of Variations for Training Set	Values/Rage of Parameters for Testing Set	No. of Variations for Testing Set
1	Sections	Sequential	Two sides of TCSC	2	Two sides of TCSC	2
2	Fault Type (10 Types).	Sequential	A-G, B-G, C-G, AB, BC, CA, AB-G, BC-G, CA-G, ABC/ABC-G	10	A-G, B-G, C-G, AB, BC, CA, AB-G, BC-G, CA-G, ABC/ABC-G	10
3	Firing angle (Compensation %)	Random	**140° to 180° (30% to 40%)**	250	136° to 180° (30% to 60%)	500
4	Fault resistance	Random	**0 Ω to 50 Ω**		0 Ω to 100 Ω	
5	Fault inception angle:	Random	0° to 360°		0° to 360°	
6	Load conditions at 5 buses	Random	90% to 110% (on each bus)		80% to 120% (on each bus)	
7	Fault Locations	Random	0 to 150 km on both sides of TCSC		0 to 150 km on both sides of TCSC	
	Total cases			**2x10x250 = 5000**		**2x10x500 = 10000**

8.4 PARAMETER TUNING

Except ELM, other SLFNs have tunable parameters associated with them. RVM has only one parameter i.e. basis width (σ) of RBF. Along with the basis width, SVM and KELM have a regularization parameter (C). Searching set of parameter-values for good performance of SLFN, which gives best (minimum) cross-validation-MSE, is vital. For all SLFNs except, ELM best values of these parameters were searched by using 5-fold cross-validation (CV) [32, 33] with nonlinear grid search [77].

For fault location, mean squared error (MSE) is used as a measure of performance of SLFN during testing and cross-validation. In this chapter, for fault location MSE for *N* number of instances (cases) is defined as

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(\text{Percent Fault Location Error}_i)^2 \tag{8.1}$$

where Percent Fault Location Error for i^{th} instance is given by:

$$\text{Percent Fault Location Error}_i = \left|\frac{\text{Predicted Fault Locaion}_i \text{ - Actual (Target) Fault Location}_i}{\text{Total Length of Transmission Line}}\right| \times 100 \tag{8.2}$$

Since ELM does not require parameter tuning, 1000 nodes were taken to make ELM-models capable of mapping required nonlinearity. Moreover, an ensemble of 100 such ELM-models was constructed to avoid overfitting by averaging out the variation in output.

8.5 RESULTS AND DISCUSSIONS

To show the efficacy of the proposed method, this section compares the performances of SLFNs with each other for the two-area system and 12-Bus system. Criteria for comparing performances are taken as MSE on output cases, mean error, maximum error, median error, cross-validation time, and training time. Based on these criteria, the most appropriate SLFN is recommended.

8.5.1 Section identification and fault-classification

Because, section identification and fault-classification are part of the proposed method and data generated in this chapter is different and more complex than previous chapter, it is again pertinent to discuss results of section identification and fault classification on this data:.

KELM has achieved best accuracy (99.41) among SLFNs for section identification (Table 8.4), while SVM gives second highest performance. Table 8.4 shows percentage accuracy of faulty section identification for different SLFNs with different type of faults on two-area system and 12-Bus system. For 12-Bus system, performance of all SLFNs have decreased. This may be due to the increased complexity of the system. However, decrease in performance of KELM and SVM is small in comparison with RVM and ELM. Additionally, Table 8.4 gives results of combined Wavelet Transform–ELM (WT-ELM) and Wavelet Transform–SVM (WT-SVM) methods reported by Malthi et el. [49] and Parekh et el. [51] respectively. Evidently, results achieved by proposed KELM based technique with modified inputs are better than results reported in [49] and [51].

Table 8.4: Section identification accuracy (%) with modified inputs.

		2 Bus System						12 Bus System			
Fault Type	**Test Cases (2 Bus / 12 Bus)**	**KELM**	**SVM**	**RVM**	**ELM**	**WT-ELM [49]**	**WT-SVM [51]**	**KELM**	**SVM**	**RVM**	**ELM**
LG	3,000	99.67	99.70	99.00	95.97	97.04	96.230	99.27	99.43	98.37	92.30
LLG	3,000	99.50	99.40	98.70	91.80	97.41	97.050	99.37	99.40	99.03	91.20
LL	3,000	99.33	99.30	97.90	91.73	95.00	88.333	98.90	96.63	96.37	90.33
LLL/LLLG	1,000	98.60	97.60	99.60	96.60	96.97	94.325	97.90	96.50	98.70	93.40
Total	**10,000**	**99.41**	**99.28**	**98.64**	**93.51**	**96.53**	**93.917**	**99.05**	**98.29**	**98.00**	**91.49**

Similar to section identification, accuracy achieved by KELM for fault classification is best in comparison with other SLFNs and results reported in [49] and [51]. This can be observed from Table 8.5 that shows percentage accuracy of fault classification with proposed approach of for all fault types.

Table 8.5: Fault classification accuracy (%) with modified inputs.

Fault Type	Test Cases (2 Bus / 12 Bus)	2-Bus System						12-BusSystem			
		KELM	SVM	RVM	ELM	WT-ELM [49]	WT-SVM [51]	KELM	SVM	RVM	ELM
LG	3,000	99.67	99.37	99.07	98.03	99.01	97.230	99.10	98.97	97.30	97.00
LLG	3,000	99.50	99.63	99.17	97.50	99.79	97.840	99.37	99.30	97.80	96.80
LL	3,000	99.27	99.10	98.63	96.50	98.25	97.29	99.03	98.80	97.10	93.33
LLL/LLLG	1,000	99.60	99.00	98.10	96.10	99.97	97.68	98.10	98.40	97.20	95.80
Total	**10,000**	**99.49**	**99.33**	**98.87**	**97.22**	**99.11**		**99.06**	**98.96**	**97.38**	**95.72**

8.5.2 Fault location

Performance of the proposed approach on testing set for fault location is mentioned in Table 8.6 with reference to various measures. In the table, it can be seen that almost on all measures KELM has given superior performance than all other SLFNs with proposed approach. Here mean squared error (MSE) and percent error are defined in (8.1) and (8.2) respectively. Maximum percent error of fault location for KELM is less than 2.68 % and 3% in case of two-area system and 12-Bus system respectively. Maximum percent fault location error for corresponding phases is shown in Table 3.1. Likewise, median of percent error is 0.214 (two-area system) and 0.230 (12-Bus system). It shows, with KELM for half of the cases error is less than 0.214% and 0.23% respectively.

On all measures, performance of 12-Bus system has slightly reduced in comparison with two-area system e.g. maximum error increased (Table 8.6, Table 8.7, and Fig. 8.4). It seems that this reduction in performance may be due to detailed modeling of the 12-Bus system.

Table 8.6: Performance of proposed approach for fault location.

	2 Bus System				12-Bus System			
SLFN	**KELM**	**SVM**	**RVM**	**ELM**	**KELM**	**SVM**	**RVM**	**ELM**
MSE (%)	0.198	0.644	3.748	7.274	0.230	0.744	4.326	8.388
Mean Error (%)	0.312	0.481	1.367	1.035	0.335	0.517	1.468	1.113
Median Error (%)	0.214	0.253	0.996	0.416	0.230	0.274	1.068	0.446
Maximum Error (%)	**2.683**	**6.952**	**14.315**	**24.584**	**2.996**	**7.848**	**14.579**	**28.219**
Training Time (s)	1.65	82.96	236.21	869.57	1.94	91.06	322.84	919.70
CV time (s)	409.38	3293.55	3851.40	--	681.57	4506.19	5056.71	--
Parameters Tuned	C, σ	C, σ	σ	None	C, σ	C, σ	σ	None
Parameter Tuning	5-Fold CV	5-Fold CV	5-Fold CV	100 Ensembles	5-Fold CV	5-Fold CV	5-Fold CV	100 Ensembles

Table 8.7: Maximum fault location error for different type of faults.

	2-Bus system				12-Bus system			
SLFN	**KELM**	**SVM**	**RVM**	**ELM**	**KELM**	**SVM**	**RVM**	**ELM**
LG	2.405	6.755	12.247	24.037	2.451	7.114	13.170	26.538
LLG	2.683	6.836	12.879	24.253	2.996	7.747	13.001	24.920
LL	2.380	6.912	10.213	24.027	2.557	7.579	11.526	26.177
LLL/LLLG	2.085	6.952	14.315	24.286	2.369	7.848	14.579	26.311
All Type of Faults	**2.683**	**6.952**	**14.315**	**24.286**	**2.996**	**7.848**	**14.579**	**26.538**

Same measures are shown with the help of boxplot in Fig. 8.4, where red line in the box shows median value, upper border of the box shows third quartile (75 percentile), and lower border of the box show first quartile (25 percentile). Furthermore, horizontal lines above and below show maximum and minimum values of percent error respectively. It can be observed from the Fig. 8.4 that except for RVM all SLFNs have error less than one percent for more than 75% cases. However, maximum error for varies significantly. Noticeably, KELM achieves least maximum error (2.68 % and 3% for two-area system and 12-Bus system respectively).

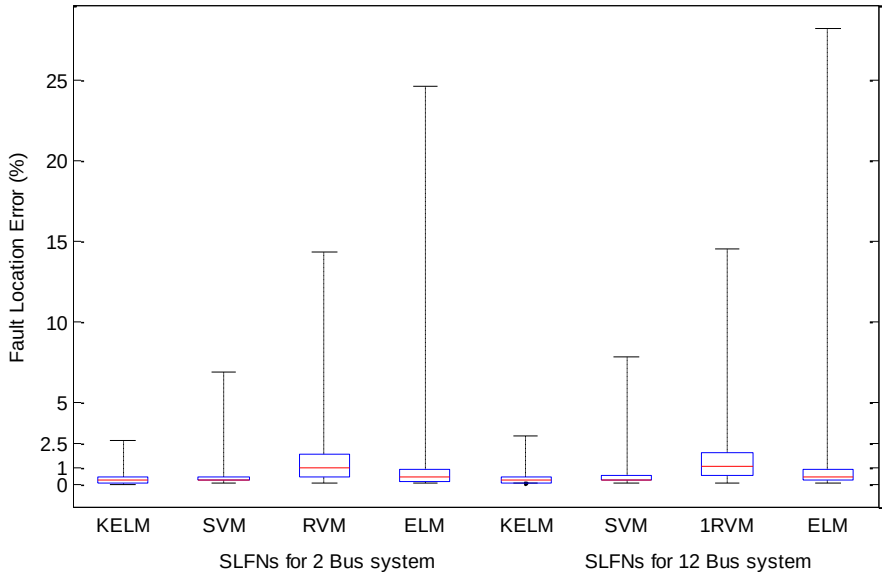


Fig. 8.4: Boxplot of percent error for fault location.

Combined time for training and cross-validation in the case of KELM is the smallest as compared to other SLFNs (seen from Table 8.6). Testing time is not shown in the table, as it is insignificant with respect to training time. Therefore, testing time is not important for fault location, which is an offline application.

8.5.3 Effect of parameter variations on fault location

From Fig. 8.5 to Fig. 8.10 show the variation of fault location errors (of KELM on testing instances) against various system and fault parameters for 2-Bus system. From these figures, it can be seen that the proposed algorithm is almost independent of variations in parameters except for fault location and fault resistance.

Moreover, the algorithm with KELM gives good results for range of parameters unseen during training (except fault resistance). For example, compensation level of 40 to 60% (i.e. firing angle 136° to 140°) was not used in training, still performance (error) for this level of compensation is similar to other range i.e. 30 to 40% (can be seen from Fig. 8.8). Similarly, for load angle (Fig. 8.9) and source impedance (Fig. 8.7), performance of KELM is not affected for unseen range of load angle and source impedance, where unseen rage of load angle is 10 to 15 degree and 25 to 30 degree, while unseen range of source impedance is 70 to 85% and 115 to 130%.

From Fig. 8.5, it can be observed that most of the cases of high errors are near to the ends of the sections of transmission line. Similarly, error of location-prediction increases with increase in fault resistance (from Fig. 8.6).

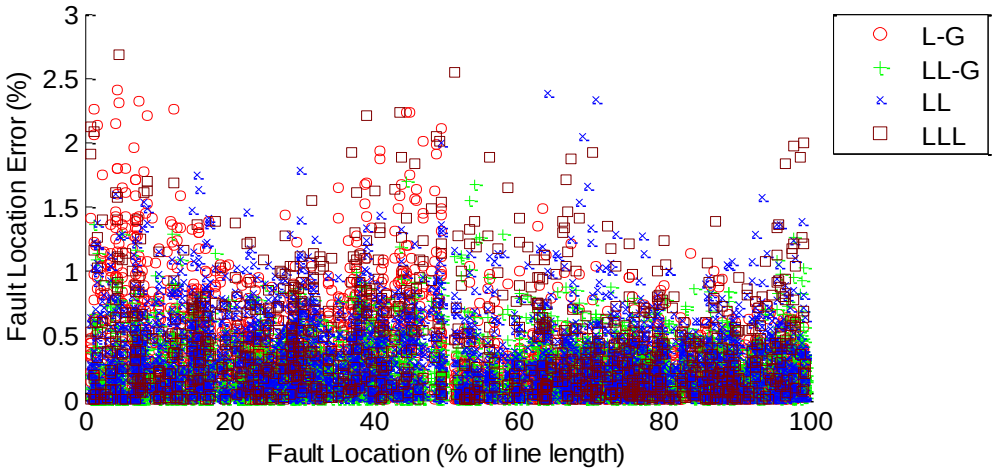


Fig. 8.5: Scatter plot of fault location error, against actual fault location.

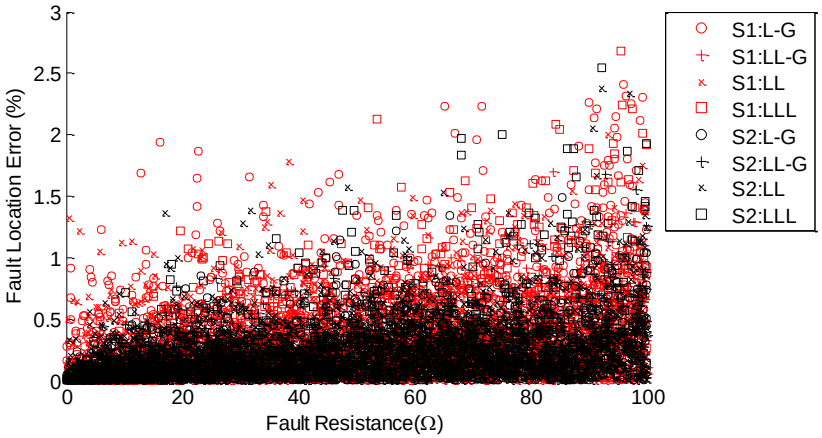


Fig. 8.6: Scatter plot of fault location error, against fault resistance (Ω).

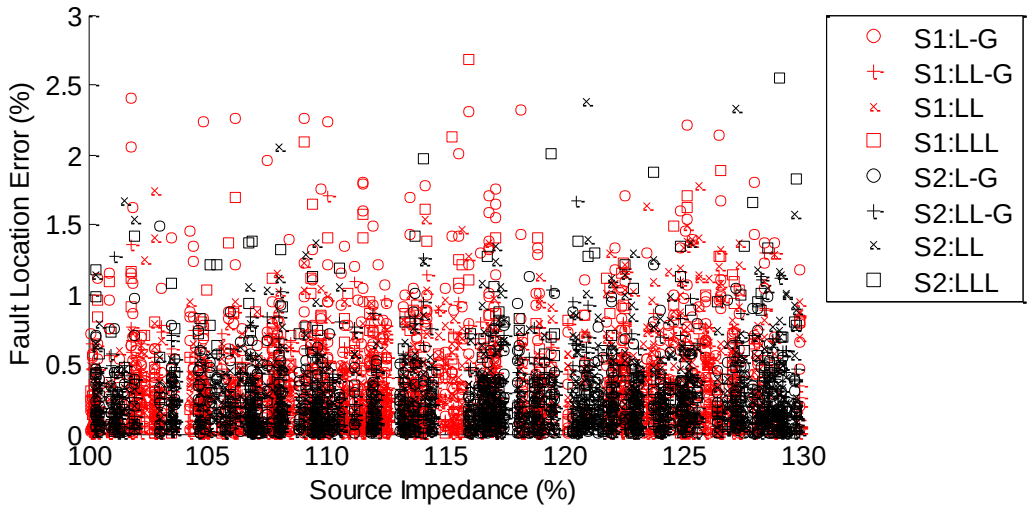


Fig. 8.7 Scatter plot of fault location error, against source impedance.

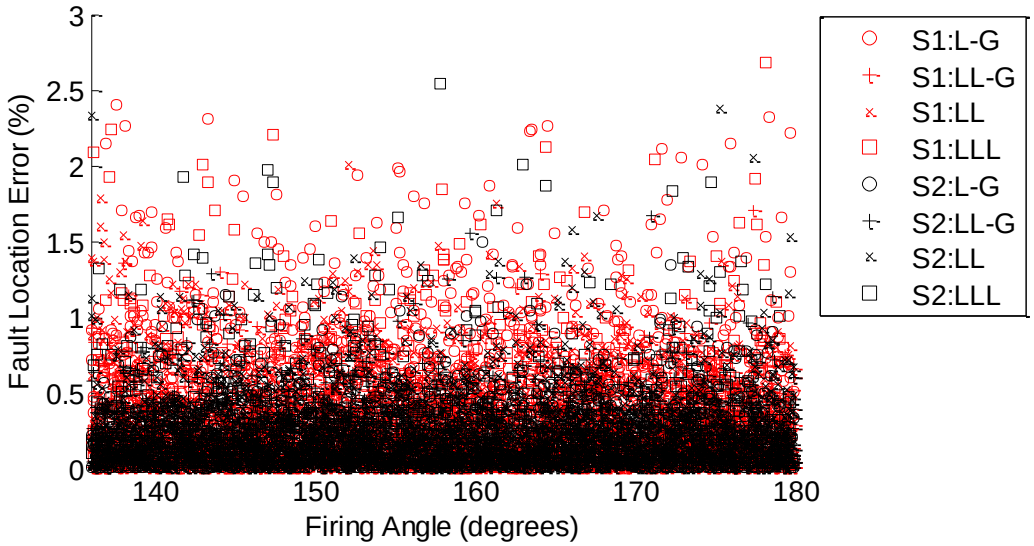


Fig. 8.8 Scatter plot of fault location error, against firing angle in degrees.

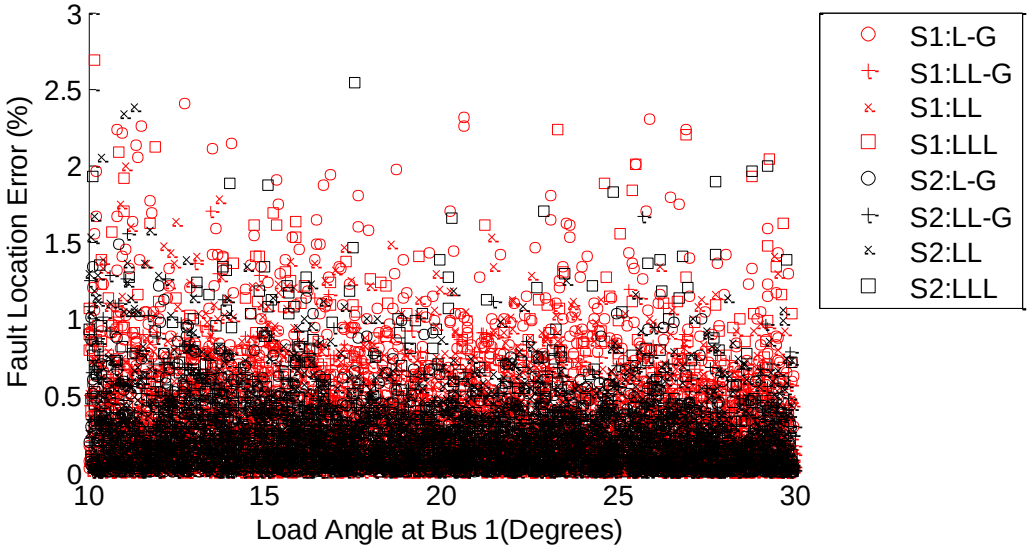


Fig. 8.9 Scatter plot of fault location error, against load angle at relaying end bus.

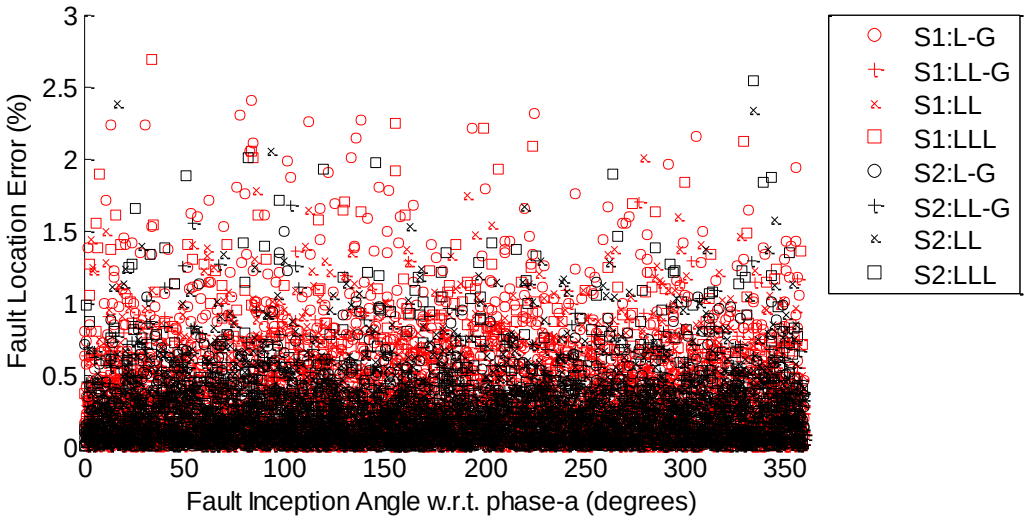


Fig. 8.10 Scatter plot of fault location error, against fault inception angle w.r.t. negative to positive zero crossing of phase-a.

8.6 CONCLUSIONS

In this chapter, a new method is proposed for fault location in series compensated line. This method used information of faulty section and fault type for modeling eight KELMs for fault location. Moreover, input sequence modification of current samples is a novelty that has resulted in improvement of accuracy. Performance of the proposed method was analyzed on a two-area system and 12-Bus system. From this analysis, KELM was found to be superior to SVM, RVM, and ELM. However, performance of SVM was very close to KELM.

Effect of variation of different system and fault parameters was analyzed on performance of KELM. Performance of KELM was found to be unaffected by variation in parameters (except fault location and fault resistance..

9.1 GENERAL CONCLUSIONS

Series compensation plays a vital role in a power system by efficiently utilizing existing infrastructure of the transmission network. However, benefits of series compensation bring with them challenges for the protection of the compensated transmission line. Protection of transmission line compensated by a fixed capacitor is inherently challenging due to conventional problems such as voltage reversal, current reversal, and nonlinear behavior of protective equipment of the capacitor. The challenges increase further with the introduction of TCSC, because of variable level of compensation and various operating modes of TCSC. This thesis is an attempt to address a few of these challenges, by predicting the location of fault, identifying type, and identifying section of the fault.

Since problem at hand is highly nonlinear, addressing it needs a tool that can map these nonlinearities. Single-hidden layer feedforward networks (SLFN) are a class of networks, which are attracting wide attention due to their capabilities of mapping high level of nonlinearities. Therefore, to address these problems, the thesis has used four popular SLFNs, namely support vector machine (SVM), relevance vector machine (RVM), extreme learning machine (ELM), and kernel extreme learning machine (KELM). Moreover, the thesis has given a brief introduction of these SLFNs, so that they can be used efficiently for the intended purpose.

Efficacies of using the SLFNs for fault classification, faulty section identification, and fault location was demonstrated on a two-area system and a 12-Bus system, simulated in PSCAD. Like any machine learning algorithms, the SLFNs require data to create, validate, and test a model. Therefore, data of fault current were created for the two-area system and the 12-Bus system by simulating multiple scenarios by taking various combinations of system and fault parameters.

Constructions of all the SLFN-models are similar. However, training process and size of the models are highly dissimilar. Accordingly, their training and testing time are also very different. Consequently, the SLFNs have different advantages and performances. Depending upon the advantages and performances, different SLFNs are appropriate for different application. Therefore, the thesis has tried to ascertain the appropriateness of the SLFNs, based on their performances and type of application such as real-time or non-real-time. For example, fault classification and faulty section identification can be used either for real-time or non-real-time purpose, whereas fault location is a non-real time application. Hence, the thesis has analyzed performances of the SLFNs and advised their suitability for fault classification, location, and faulty section identification.

9.2 MAIN CONCLUSIONS

1. The thesis has applied RVM and SVM for proposing new methods for fault classification in series compensated transmission lines. The optimal values of parameters of RVM and SVM were searched using cross-validation. Genetic algorithm (GA) was used to search parameters of SVM efficiently. Optimum values of sampling rate, data window length, and size of training set were decided by analyzing results of a large number (792) of experiments. Half cycle post-fault three phases current samples sampled at 4 kHz were found to give the best results. Performance of RVM and SVM were compared based on execution time and classification accuracy. Very high fault classification accuracy (more than 99.50%) was achieved by RVM as well as SVM.

   Therefore, a conclusion can be drawn that RVM is the most suitable SLFN for real-time application, because it has very small execution time and storage requirements.

2. Further, the thesis has given a new method for faulty section identification by employing the four popular SLFNs i.e. SVM, RVM, ELM, and KELM. Implementing of the SLFNs on two-area system and the 12-Bus system was used to demonstrate efficacies of SLFNs. The results of these implementations were analyzed on the scale of accuracy, training time, storage requirements, and prediction time. From this analysis, different SLFNs were recommended for real-time and non-real-time applications.

   RVM was found fit for applications of faulty section identification that require minimal storage and fast prediction. While the applications that require repeated training and highest accuracy, found KELM and SVM appropriate for them.

   Although, ELM does not need parameter-tuning, it could not be suggested either for real-time or non-real-time application, due to its large prediction time and poor accuracy on complex system such as the 12-Bus system.

3. The thesis has provided a new two-step method for locating faults in the series compensated lines. In this method first step of faulty section identification assists, the second step of locating faults. Regression analysis was performed with the help of the four SLFNs to predict the fault location. Implementation on the two-area system and the 12-Bus system has established the effectiveness of the SLFNs for fault location.

   Among the four SLFNs, KELM was most attractive for fault location, which is non-real-time application, because KELM achieved very small MSE (comparable to SVM) and the smallest maximum-fault-location-error. Moreover, in comparison with

RVM and SVM, KELM has taken very small overall time that included time for tuning parameter, training, and prediction of fault location.

4. A second method for fault location is proposed, in which sequence of input samples was modified according to type of fault (L-G, LL-G, LL, and LLL-G) to attain enhanced accuracy. Effect of parameter variation on this method was analyzed. Again, in this method, KELM performed better than other SLFNs.
5. The thesis has proposed two innovative approaches to enhance the batch mode capabilities of existing emtp-type program. Batch mode enables simulation of a large number of scenarios with different parameter settings. Although the proposed approaches were used to record current signals during faults, it can be concluded that these approaches can be used for any application that requires repeated simulations with different settings.
6. In the first approach, minor modifications are made to the program structure of a popular emtp-type program (PSCAD). This allows changing of the transmission line length in batch mode. This saved nearly two person-weeks for a case studied on a two-area power system model.
7. Second generalized two-step approach was proposed to change the values of any parameters, because the first approach was not capable to alter the values of many parameters other than transmission line length. More time and efforts were saved by this approach for generating data on the two-area system. Additionally, this approach has saved more than one person-months for the 12-Bus system.

   Besides, the generalized nature of the approach opens possibilities for its use in other applications where a repeated change of values is required such as searching optimal value of a parameter.
8. Overall, this thesis has taken humble strides to competently classify a fault, identify section of the fault, and locate the fault in the series compensated transmission lines.

PUBLICATIONS FROM THE WORK

Journals - Published/Communicated

[1] Pushkar Tripathi and H. O. Gupta, "A method for fast and accurate fault classification in TCSC compensated transmission line using RVM," International Review of Electrical Engineering, vol. 8, pp. 1608-1615, 2013. *(Published)*

[2] Pushkar Tripathi, G. N. Pillai, and H. O. Gupta, "Kernel-ELM based Fault Location in Advanced Series-Compensated Line," Electric Power Components and Systems, Taylor & Francis. *(Communicated after revision).*

International conferences

[1] Pushkar Tripathi, G. N. Pillai, and H. O. Gupta, "New method for Fault Classification in TCSC Compensated Transmission Line using GA Tuned SVM," presented at the POWERCON 2012, New Zealand, Auckland, 2012.

[2] Pushkar Tripathi, G. N. Pillai, H. O. Gupta, and A. Kumar, "PSCAD's Capability Enhancement for Transmission Line Fault Diagnosis by Changing Fault Location Automatically," presented at the IEEE - International Conference on Research and Development Prospectus on Engineering and Technology (ICRDPET 2013), Nagapattinam, Tamil Nadu, India, 2013.

[3] Pushkar Tripathi, G. N. Pillai, H. O. Gupta, and A. Kumar, "Externally modifying parameters and invoking electromagnetic transients simulation in PSCAD," presented at the Power Electronics, Drives and Energy Systems (PEDES), 2014 IEEE International Conference on December 2014, at IIT Bombay, India.

BIBLIOGRAPHY

[1] N. G. Hingorani, L. Gyugyi, and M. El-Hawary, *Understanding FACTS: concepts and technology of flexible AC transmission systems* vol. 1: IEEE press New York, 2000.

[2] R. Mohan Mathur and R. K. Varma, *Thyristor-Based Facts Controllers for Electrical Transmission Systems*: John Wiley & Sons, inc. Publication, 2002.

[3] J. G. Singh, S. N. Singh, and S. C. Srivastava, "Enhancement of power system security through optimal placement of TCSC and UPFC," pp. 1-6.

[4] R. A. Castro and H. A. Pineda, "Protection System Considerations for 400 kV Series Compensated Transmission Lines of the Central Western Network in Venezuela," in *Transmission & Distribution Conference and Exposition: Latin America, 2006. TDC '06. IEEE/PES*, 2006, pp. 1-5.

[5] M. Tripathy and S. Mishra, "Interval type-2-based thyristor controlled series capacitor to improve power system stability," *Generation, Transmission & Distribution, IET,* vol. 5, pp. 209-222, 2011.

[6] P. K. Dash, S. Mishra, and G. Panda, "Damping multimodal power system oscillation using a hybrid fuzzy controller for series connected FACTS devices," *Power Systems, IEEE Transactions on,* vol. 15, pp. 1360-1366, 2000.

[7] B. K. Kumar, S. N. Singh, and S. C. Srivastava, "A modal controllability index for optimal placement of TCSC to damp inter-area oscillations," in *Power Engineering Society General Meeting, 2005. IEEE*, 2005, pp. 1653-1657 Vol. 2.

[8] R. K. Varma, S. Auddy, and Y. Semsedini, "Mitigation of subsynchronous resonance in a series-compensated wind farm using FACTS controllers," *Power Delivery, IEEE Transactions on,* vol. 23, pp. 1645-1654, 2008.

[9] M. Khederzadeh and T. S. Sidhu, "Impact of TCSC on the protection of transmission lines," *Power Delivery, IEEE Transactions on,* vol. 21, pp. 80-87, 2006.

[10] W. Wang, X. Yin, J. Yu, X. Duan, and D. Chen, "The impact of TCSC on distance protection relay," in *Power System Technology, 1998. Proceedings. POWERCON '98. 1998 International Conference on*, 1998, pp. 382-388 vol.1.

[11] S. Wilkinson, "Series Compensated Line Protection Issues," *GE Publication GER-3972 (www.multilin.com),* 1999.

[12] H. J. Altuve, J. B. Mooney, and G. E. Alexander, "Advances in series-compensated line protection," in *Protective Relay Engineers, 2009 62nd Annual Conference for*, 2009, pp. 263-275.

[13] Q. Y. Xuan and A. T. Johns, "Digital simulation of series-compensated EHV (extra high voltage) transmission systems," in *Simulation of Power Systems, IEE Colloquium on*, 1992, pp. 8/1 -8/3.

[14] "Single phase tripping and auto reclosing of transmission lines-IEEE Committee Report," *Power Delivery, IEEE Transactions on,* vol. 7, pp. 182-192, Jan 1992.

[15] A. A. Girgis, A. A. Sallam, and A. K. El-Din, "An adaptive protection scheme for advanced series compensated (ASC) transmission lines," *Power Delivery, IEEE Transactions on,* vol. 13, pp. 414 -420, apr 1998.

[16] C. Aguilera, E. Orduna, and G. Ratta, "Fault detection, classification and faulted phase selection approach based on high-frequency voltage signals applied to a series-compensated line," *Generation, Transmission and Distribution, IEE Proceedings-,* vol. 153, pp. 469-475, 2006.

[17] B. Bhalja and R. P. Maheshwari, "Trends in Adaptive Distance Protection of Multiterminal and Double-Circuit Lines," *Electric Power Components and Systems,* vol. 34, pp. 603-617, 2006/06/01 2006.

[18] B. Das and J. V. Reddy, "Fuzzy-logic-based fault classification scheme for digital distance protection," *Power Delivery, IEEE Transactions on,* vol. 20, pp. 609-616, 2005.

[19] A. K. Pradhan, A. Routray, and B. Biswal, "Higher order statistics-fuzzy integrated scheme for fault classification of a series-compensated transmission line," *Power Delivery, IEEE Transactions on,* vol. 19, pp. 891 - 893, april 2004.

[20] A. K. Pradhan, A. Routray, S. Pati, and D. K. Pradhan, "Wavelet fuzzy combined approach for fault classification of a series-compensated transmission line," *Power Delivery, IEEE Transactions on,* vol. 19, pp. 1612 - 1618, oct. 2004.

[21] D. W. P. Thomas and C. Christopoulos, "Ultra-high speed protection of series compensated lines," *Power Delivery, IEEE Transactions on,* vol. 7, pp. 139 -145, jan 1992.

[22] Z. Moravej, M. Pazoki, and A. A. Abdoos, "A new approach for fault classification and section detection in compensated transmission line with TCSC," *European Transactions on Electrical Power,* vol. 21, pp. 997-1014, 2011.

[23] Y. H. Song, A. T. Johns, and Q. Y. Xuan, "Radial basis function neural networks for fault diagnosis in controllable series compensated transmission lines," in *Electrotechnical Conference, 1996. MELECON '96., 8th Mediterranean*, 1996, pp. 1449 -1452 vol.3.

[24] W. J. Cheong and R. K. Aggarwal, "A novel feature extraction and optimisation method for neural network-based fault classification in TCSC-compensated lines," in *Power Engineering Society Summer Meeting, 2002 IEEE* vol. 2, ed, 2002, pp. 795 -800 vol.2.

[25] Q. Y. Xuan, Y. H. Song, A. T. Johns, R. Morgan, and D. Williams, "Performance of an adaptive protection scheme for series compensated EHV transmission systems using neural networks," *Electric Power Systems Research,* vol. 36, pp. 57-66, 1996.

[26] S. R. Samantaray and P. K. Dash, "Pattern recognition based digital relaying for advanced series compensated line," *International Journal of Electrical Power & Energy Systems,* vol. 30, pp. 102-112, 2// 2008.

[27] Y. H. Song, A. T. Johns, Q. Y. Xuan, and J. Y. Liu, "Genetic algorithm based neural networks applied to fault classification for EHV transmission lines with a UPFC," in *Developments in Power System Protection, Sixth International Conference on (Conf. Publ. No. 434)*, 1997, pp. 278-281.

[28] P. K. Dash, S. R. Samantaray, and G. Panda, "Fault Classification and Section Identification of an Advanced Series-Compensated Transmission Line Using Support Vector Machine," *Power Delivery, IEEE Transactions on,* vol. 22, pp. 67 -73, jan. 2007.

[29] U. B. Parikh, B. Das, and R. Maheshwari, "Fault classification technique for series compensated transmission line using support vector machine," *International Journal of Electrical Power & Energy Systems,* vol. 32, pp. 629-636, 2010.

[30] B. Bhalja and R. P. Maheshwari, "Wavelet-based Fault Classification Scheme for a Transmission Line Using a Support Vector Machine," *Electric Power Components and Systems,* vol. 36, pp. 1017-1030, 2008/09/24 2008.

[31] T. S. Sidhu, L. Mital, and M. S. Sachdev, "A comprehensive analysis of an artificial neural-network-based fault direction discriminator," *Power Delivery, IEEE Transactions on,* vol. 19, pp. 1042-1048, 2004.

[32] H. Simon, *NEURAL NETWORKS: A Comprehensive Foundation*: Printice-Hall, Inc., 1999.

[33] C. M. Bishop, *Pattern Recognition and Machine Learning*: Springer, 2006.

[34] B. Y. Vyas, R. P. Maheshwari, and B. Das, "Improved fault analysis technique for protection of Thyristor controlled series compensated transmission line," *International Journal of Electrical Power & Energy Systems,* vol. 55, pp. 321-330, 2// 2014.

[35] P. Mazniewski and J. Izykowski, "Identification of faulted section in TCSC transmission line based on DC component measurement," in *Universities Power Engineering Conference (UPEC), 2009 Proceedings of the 44th International*, 2009, pp. 1-5.

[36] B. Vyas, R. P. Maheshwari, and B. Das, "Protection of series compensated transmission line: Issues and state of art," *Electric Power Systems Research,* vol. 107, pp. 93-108, 2014.

[37] M. G. Ahsaee and J. Sadeh, "A Novel Fault-Location Algorithm for Long Transmission Lines Compensated by Series FACTS Devices," *Power Delivery, IEEE Transactions on,* vol. 26, pp. 2299-2308, 2011.

[38] J. Sadeh, N. Hadjsaid, A. M. Ranjbar, and R. Feuillet, "Accurate fault location algorithm for series compensated transmission lines," *Power Delivery, IEEE Transactions on,* vol. 15, pp. 1027-1033, 2000.

[39] M. M. Saha, J. Izykowski, E. Rosolowski, and B. Kasztenny, "A new accurate fault locating algorithm for series compensated lines," *Power Delivery, IEEE Transactions on,* vol. 14, pp. 789-797, 1999.

[40] F. H. Magnago and A. Abur, "Fault location using wavelets," *Power Delivery, IEEE Transactions on,* vol. 13, pp. 1475-1480, 1998.

[41] S. Mishra, C. N. Bhende, and K. B. Panigrahi, "Detection and classification of power quality disturbances using S-transform and probabilistic neural network," *Power Delivery, IEEE Transactions on,* vol. 23, pp. 280-287, 2008.

[42] J. Mora-Florez, V. Barrera-Nuez, and G. Carrillo-Caicedo, "Fault Location in Power Distribution Systems Using a Learning Algorithm for Multivariable Data Analysis," *Power Delivery, IEEE Transactions on,* vol. 22, pp. 1715-1721, 2007.

[43] V. Malathi, N. S. Marimuthu, and S. Baskar, "Intelligent approaches using support vector machine and extreme learning machine for transmission line protection," *Neurocomputing,* vol. 73, pp. 2160-2167, 6// 2010.

[44] M. E. Tipping, "Sparse Bayesian learning and the relevance vector machine," *The journal of machine learning research,* vol. 1, pp. 211-244, 2001.

[45] M. E. Tipping, "The Relevance Vector Machine," presented at the Advances in Neural Information Processing Systems, 2000.

[46] X.-m. Xu, Y.-f. Mao, J.-n. Xiong, and F.-l. Zhou, "Classification Performance Comparison between RVM and SVM," in *Anti-counterfeiting, Security, Identification, 2007 IEEE International Workshop on*, 2007, pp. 208-211.

[47] G.-B. Huang, D. H. Wang, and Y. Lan, "Extreme learning machines: a survey," *International Journal of Machine Learning and Cybernetics,* vol. 2, pp. 107-122, 2011.

[48] G.-B. Huang, Q.-Y. Zhu, and C.-K. Siew, "Extreme learning machine: theory and applications," *Neurocomputing,* vol. 70, pp. 489-501, 2006.

[49] V. Malathi, N. S. Marimuthu, S. Baskar, and K. Ramar, "Application of extreme learning machine for series compensated transmission line protection," *Engineering Applications of Artificial Intelligence,* vol. 24, pp. 880-887, 8// 2011.

[50] H. Guang-Bin, Z. Hongming, D. Xiaojian, and Z. Rui, "Extreme Learning Machine for Regression and Multiclass Classification," *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on,* vol. 42, pp. 513-529, 2012.

[51] U. B. Parikh, B. Das, and R. P. Maheshwari, "Combined Wavelet-SVM Technique for Fault Zone Detection in a Series Compensated Transmission Line," *Power Delivery, IEEE Transactions on,* vol. 23, pp. 1789-1794, 2008.

[52] "PSCAD/EMTDC," X4 (4.3) ed: Manitoba HVDC Research center.

[53] P. Vuorenpää, P. Järventausta, and J. Lavapuro, "Dynamic Modeling of Thyristor Controlled Series Capacitor in PSCAD and RTDS Environments," presented at the Nordic Workshop on Power and Industrial Electronics, , 2008.

[54] T. S. Sidhu and M. Khederzadeh, "TCSC impact on communication-aided distance-protection schemes and its mitigation," *IEE Proceedings - Generation, Transmission and Distribution,* vol. 152, pp. 714-728, 2005.

[55] Q. Liu, Z. Wang, and Y. Xu, "Study on the Influence of TCSC on Fault Component Distance Protection," in *Transmission and Distribution Conference and Exhibition: Asia and Pacific, 2005 IEEE/PES*, 2005, pp. 1-4.

[56] S. Kolla and L. Varatharasa, "Identifying three-phase induction motor faults using artificial neural networks," *ISA transactions,* vol. 39, pp. 433-439, 2000.

[57] L. Mohan Saini and M. K. Soni, "Artificial neural network-based peak load forecasting using conjugate gradient methods," *Power Systems, IEEE Transactions on,* vol. 17, pp. 907-912, 2002.

[58] D. K. Chaturvedi, O. P. Malik, and P. K. Kalra, "Experimental studies with a generalized neuron-based power system stabilizer," *Power Systems, IEEE Transactions on,* vol. 19, pp. 1445-1453, 2004.

[59] M. S. Ballal, H. M. Suryawanshi, and M. K. Mishra, "ANN Based System for the Detection of Winding Insulation Condition and Bearing Wear in Single Phase Induction Motor," *Journal of Electrical Engineering & Technology,* vol. 2, pp. 485-493, 2007.

[60] M. S. Ballal, H. M. Suryawanshi, and M. K. Mishra, "ANN based real time incipient fault detection and protection system for induction motor," *International Journal of Power and Energy Conversion,* vol. 1, pp. 125-142, 2009.

[61] J. M. Zurada, *Introduction to artificial neural systems* vol. 8: West publishing company St. Paul, 1992.

[62] C. J. C. Burges, "A Tutorial on Support Vector Machines for Pattern Recognition," *Data Mining and Knowledge Discovery,* vol. 2, pp. 121--167, 1998.

[63] S. K. Aggarwal, L. M. Saini, and A. Kumar, "Day-ahead price forecasting in Ontario electricity market using variable-segmented support vector machine-based model," *Electric Power Components and Systems,* vol. 37, pp. 495-516, 2009.

[64] K. Seethalekshmi, S. N. Singh, and S. C. Srivastava, "A Classification Approach Using Support Vector Machines to Prevent Distance Relay Maloperation Under Power Swing and Voltage Instability," *Power Delivery, IEEE Transactions on,* vol. 27, pp. 1124-1133, 2012.

[65] P. Janik, T. Lobos, and P. Schegner, "Classification of power quality events using SVM networks," in *Developments in Power System Protection, 2004. Eighth IEE International Conference on*, 2004, pp. 768-771 Vol.2.

[66] P. Janik and T. Lobos, "Automated classification of power-quality disturbances using SVM and RBF networks," *Power Delivery, IEEE Transactions on,* vol. 21, pp. 1663-1669, 2006.

[67] S. R. Kolla, "Identifying faults in three phase induction motors using support vector machines," in *proc. Electrical manufacturing and coil winding expo*, 2010-2013, pp. 109-114.

[68] A. Tolambiya and P. K. Kalra, "Relevance vector machine with adaptive wavelet kernels for efficient image coding," *Neurocomputing,* vol. 73, pp. 1417-1424, 3// 2010.

[69] G.-B. Huang and C.-K. Siew, "Extreme learning machine with randomly assigned RBF kernels," *International Journal of Information Technology,* vol. 11, pp. 16-24, 2005.

[70] C. Cortes and V. Vapnik, "Support-Vector Networks," *Machine Learning,* vol. 20, pp. 273-297, 1995.

[71] C. Chang and C. Lin, "LIBSVM: A library for support vector machines," *ACM Trans. Intell. Syst. Technol.,* vol. 2, May// 2011.

[72] V. Vapnik, S. E. Golowich, and A. Smola, "Support Vector Method for Function Approximation, Regression Estimation, and Signal Processing," in *Advances in Neural Information Processing Systems 9* M. M., J. M., and P. T., Eds., ed: MIT Press, 1996, pp. 281--287.

[73] M.E.Tipping. (2009). *SPARSEBAYES Matlab Toolbox (2.00 ed.)*. Available: http://www.relevancevector.com

[74] D. MacKay, "The Evidence Framework Applied to Classification Networks," *Neural Computation,* vol. 4, pp. 720-736, 1992.

[75] P. L. Bartlett, "The sample complexity of pattern classification with neural networks: the size of the weights is more important than the size of the network," *Information Theory, IEEE Transactions on,* vol. 44, pp. 525-536, 1998.

[76] H.-M. Zhou and G.-B. Huang. (2012, 11 February 2014). ELM with Kernels. Available: http://www.ntu.edu.sg/home/egbhuang/elm_codes.html

[77] C.-W. Hsu, C.-C. Chang, and C.-J. Lin, "A practical guide to support vector classification," ed, 2003.

[78] C.-W. Hsu and C.-J. Lin, "A comparison of methods for multiclass support vector machines," *Neural Networks, IEEE Transactions on,* vol. 13, pp. 415 -425, mar 2002.

[79] V. N. Vapnik, "An overview of statistical learning theory," *Neural Networks, IEEE Transactions on,* vol. 10, pp. 988 -999, sep 1999.

[80] N. Liu and H. Wang, "Ensemble based extreme learning machine," *Signal Processing Letters, IEEE,* vol. 17, pp. 754-757, 2010.

[81] W. Deng, Q. Zheng, and L. Chen, "Regularized extreme learning machine," pp. 389-395.

[82] A. M. Gole, O. B. Nayak, T. S. Sidhu, and M. S. Sachdev, "A graphical electromagnetic simulation laboratory for power systems engineering programs," *Power Systems, IEEE Transactions on,* vol. 11, pp. 599-606, 1996.

[83] J. Mahseredjian, V. Dinavahi, and J. A. Martinez, "Simulation Tools for Electromagnetic Transients in Power Systems: Overview and Challenges," *Power Delivery, IEEE Transactions on,* vol. 24, pp. 1657-1669, 2009.

[84] J. Mahseredjian, V. Dinavahi, and J. A. Martinez, "An Overview of Simulation Tools for Electromagnetic Transients in Power Systems," in *Power Engineering Society General Meeting, 2007. IEEE*, 2007, pp. 1-6.

[85] *USER'S GUIDE: A Comprehensive Resource for EMTDC*: Manitoba HVDC Research Centre, 2010.

[86] J. Shan, U. D. Annakkage, and A. M. Gole, "A platform for validation of FACTS models," *Power Delivery, IEEE Transactions on,* vol. 21, pp. 484-491, 2006.

[87] H. W. Dommel, "Digital Computer Solution of Electromagnetic Transients in Single-and Multiphase Networks," *Power Apparatus and Systems, IEEE Transactions on,* vol. PAS-88, pp. 388-399, 1969.

[88] H. W. Dommel and N. Sato, "Fast Transient Stability Soultions," *Power Apparatus and Systems, IEEE Transactions on,* vol. PAS-91, pp. 1643-1650, 1972.

[89] A. M. Gole, S. Filizadeh, R. W. Menzies, and P. L. Wilson, "Optimization-enabled electromagnetic transient simulation," *Power Delivery, IEEE Transactions on,* vol. 20, pp. 512-518, 2005.

[90] O. Nayak, G. Irwin, and A. Neufeld, "GUI enhances electromagnetic transients simulation tools," *Computer Applications in Power, IEEE,* vol. 8, pp. 17-22, 1995.

[91] P. Tripathi, G. N. Pillai, and H. O. Gupta, "New method for Fault Classification in TCSC Compensated Transmission Line using GA Tuned SVM," presented at the POWERCON 2012, Newzealand, Auckland, 2012.

[92] (2012, 01-December). *AutoIt (3.3.8.1 ed.)*. Available: http://www.autoitscript.com/autoit3/

[93] (2012, 01-December). *AutoHotkey (1.1.09.02 ed.)*. Available: http://www.autohotkey.com/

[94] S. Filizadeh and A. M. Gole, "Optimal design of power electronic systems using electromagnetic transient simulation," in *Electrical and Computer Engineering, 2005. Canadian Conference on*, 2005, pp. 450-453.

[95] P. Tripathi, A. Sharma, G. N. Pillai, and I. Gupta, "Accurate Fault Classification and Section Identification Scheme in TCSC Compensated Transmission Line using SVM," *World Academy of Science, Engineering and Technology,* vol. 60, pp. 1599-1605, 2011.

[96] R. D. Zimmerman, S. Murillo, x, C. E. nchez, and R. J. Thomas, "MATPOWER: Steady-State Operations, Planning, and Analysis Tools for Power Systems Research and Education," *Power Systems, IEEE Transactions on,* vol. 26, pp. 12-19, 2011.

[97] (2012, 17-Jan). *Global Optimization: To find minimum of function using GA*. Available: http://www.mathworks.in/help/toolbox/gads/ga.html

[98] D. J. Lawrence and D. L. Waser, "Transmission line fault location using digital fault recorders," *Power Delivery, IEEE Transactions on,* vol. 3, pp. 496-502, 1988.

[99] M. Chantler, P. Pogliano, A. Aldea, G. Tornielli, T. Wyatt, and A. Jolley, "The use of fault-recorder data for diagnosing timing and other related faults in electricity transmission networks," *Power Systems, IEEE Transactions on,* vol. 15, pp. 1388-1393, 2000.

[100] M. Kezunovic, "Smart Fault Location for Smart Grids," *Smart Grid, IEEE Transactions on,* vol. 2, pp. 11-22, 2011.

[101] A. G. Phadke and J. S. Thorp, *Computer relaying for power systems*: John Wiley & Sons, 2009.

[102] M. M. Saha, J. Izykowski, E. Rosolowski, P. Balcerek, and M. Fulczyk, "Accurate location of faults on series-compensated lines with use of two-end unsynchronised measurements," *IET Conference Proceedings,* pp. 338-343, 2008.

A1. System Parameters for two-area system

(i) Source 1 and 2:

Positive sequence impedance = $15.06\angle 85^{o}\Omega$ (100%)

Zero sequence impedance = $26.70\angle 85^{o}\Omega$ (100%)

Frequency = 50 Hz

Voltage at terminal = 400 kV.

Phase angle= varied between 10° to 30° for source 1 and kept 0° for source 2.

(ii) Transmission-Line Data:

Length = 300 km

Voltage = 400 kV

Positive-sequence impedance = 8.212+j83.286 Ω

Zero-sequence impedance = 94.343+j250.667 Ω

Positive-sequence admittance = 3+j1241 µ-mho

Zero-sequence admittance = 3+j833.83 µ-mho

A2. Specifications of machine (for fault classification):

Intel® Core™2 CPU, E7400@2.80 GHz, 4 GB RAM.

Microsoft Windows, XP Professional 2002, Service Pack 2.

A3. Specifications of machine (for section Identification and fault location):

Intel® Core™ i7-2600 CPU @ 3.40GHz

16.0 GB Ram, 64-bit Windows 7