AN INVESTIGATION ON
DEEP LEARNING AND MULTI-LABEL LEARNING
FOR COMPOSITE SYSTEM RELIABILITY EVALUATION

A Dissertation

by

DOGAN URGUN

Submitted to the Office of Graduate and Professional Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Chair of Committee,	Chanan Singh
Committee Members,	Methrad Ehsani
	Alex Spritson
	Sergey Butenko
Head of Department,	Miroslav Begovic

May 2019

Major Subject: Electrical Engineering

ABSTRACT

In many cases, research on reliability analysis focuses on searching the state space of the system for states that represent events of interest, like failure of the system not meeting the required demand for a specific node. This raises the need for search procedures that efficiently determine states to be examined and then evaluated. Artificial Intelligence based methods have been studied for this objective either by themselves or in conjunction with widely used methods like Monte Carlo Simulation. This dissertation investigates various novel approaches for reliability evaluation of composite power systems by combining Monte Carlo simulation (MCS) with different machine learning techniques for Multi-Label Learning and Deep Learning topologies. The objective in this research is reducing the computational burden to perform Monte Carlo Simulation for a given level of accuracy. As a consequence, higher accuracy can be obtained for the same level of computational effort.

To my family and beloved wife

ACKNOWLEDGEMENTS

Many people deserve special acknowledgment for their support throughout the duration of this work.

First, I must acknowledge my deep gratitude to my advisor, Dr. Chanan Singh, for his excellent professional guidance and constant assistance rendered in making this dissertation possible. His insightful comments and suggestions fueled the research. I want to say that working under his supervision during my Ph.D. study is one of the luckiest things in my life. His mentoring will benefit me long term.

Next, I would like to express great thanks to Dr. Methrad Ehsani, Dr. Alex Spritson, and Dr. Sergey Butenko for their effort in serving as members of my Ph.D. studies committee. They consistently provided strong support to my career development in the academic environment. I really appreciate their help during these years.

I am indebted to many friends for their assistance and advice. They have stood by and encouraged me when my productivity waned. I also wish to thank my family and my beloved wife for their ever-present encouragement and for the moral and practical support given over the years before and throughout this endeavor. I dedicate this dissertation to them.

CONTRIBUTORS AND FUNDING SOURCES

Contributors

This work was supported by a dissertation committee consisting of Professor Chanan Singh (Advisor) and Professors Methrad Ehsani and Alex Sprintson of the Electrical Engineering Department and Professor Sergey Butenko of the Industrial Engineering Department. All the work for this dissertation was independently completed by the student.

Funding Sources

The research in this dissertation was partly supported by PSERC Project # S-75: Reliability Evaluation of Renewable Generation Integrated Power Grid including Adequacy and Dynamic Security Assessment.

TABLE OF CONTENTS

Page

ABSTRACT ii

DEDICATION iii

ACKNOWLEDGEMENTS iv

CONTRIBUTORS AND FUNDING SOURCES....v

TABLE OF CONTENTS vi

LIST OF FIGURES....ix

LIST OF TABLES x

CHAPTER I INTRODUCTION 1

A.Introduction 1
B. Research Objectives 3
C.Dissertation Structure 3

CHAPTER II CONCEPT OF COMPOSITE POWER SYSTEM RELIABILITY EVALUATION AND MONTE CARLO SIMULATION....5

A.Monte Carlo Simulation 5
1.Non-Sequential Monte Carlo Simulation....7
2.Sequential Monte Carlo Simulation 8
a.Fixed Time Interval Method....8
b.Next Event Method 9
B.Composite System Adequacy Analysis....10
1.DC Power Model....11
2. AC Power Model....12
C.Reliability Test Systems....13
1.IEEE 30 Bus Test System 14
2. IEEE RTS 79 Test System 15
D.Performance Evaluation Metrics....16

CHAPTER III MULTI-LABEL CLASSIFICATION FOR COMPOSITE POWER SYSTEM RELIABILITY EVALUATION 18

A.Multi-Label Learning for Power System Reliability Evaluation 19
B. Multi-Label K-Nearest Neighbor Algorithm...20
C. MLKNN for Power System Reliability Evaluation..22
1. General Definition of MLKNN Parameters 23
2. Explanation of MLKNNN Procedure..24
D. Proposed Topology...29
E. Case Studies and Results 33
1. Performance on IEEE 30 Bus Test System 34
2. Performance on IEEE RTS..35
3. Performance on Varying Hourly Load 39
4. Performance Including Transmission Line Failures 42
F. Conclusion 45

CHAPTER IV MULTI-LABEL CLASSIFICATION AND IMPORTANCE SAMPLING COMBINATION FOR COMPOSITE SYSTEM RELIABILITY EVALUATION...48

A. MLRBF Classification for Power System Reliability Evaluation..........................48
1. General Definitions for MLRBF in Power System Reliability Evaluation..........49
2. Explanation of MLRBF Classification Procedure...50
B. Importance Sampling..53
1. General Definition of Importance Sampling 54
2. CE Algorithm 55
C. Application Procedure of Proposed Method 58
1.Training Process 58
2.Testing Process...59
D. Case Studies and Results..60
1. Constant Load Level 61
2. Hourly Varying Load Levels..63
E.Conclusion 66

CHAPTER V DEEP LEARNING FOR COMPOSITE POWER SYSTEM RELIABILITY EVALUATION 67

A. Convolutional Neural Networks...68
1. Convolutional Layer...69
2.Rectified Linear Unit(ReLU)..70
3.Pooling Layer 70
4.Dropout Technique..71
5.Fully Connected Layer 72
B. Implementation of CNN in Composite System Reliability Evaluation...................72
1. Creating Balanced Datasets...73
a. Size Issue in Training Datasets..73
b. Class Imbalance Problem 74

c. Proposed Solution........75
2. CNN Architectures for the Proposed Method........77
a. Designed CNN Architecture........78
3. Testing Stage........80
C. Case Studies and Results........82
1. Constant Load Level........82
2. Varying Hourly Load Level........84
D. Conclusion........87

CHAPTER VI CONCLUDING REMARKS........89

A. Conclusion........89
B. Future Work and Suggestions........91

REFERENCES........93

LIST OF FIGURES

Page

Figure 2.1: Schematic for IEEE 30 Bus Test System. 14

Figure 2.2: Schematic for IEEE RTS 15

Figure 3.1: Overall Diagram for MLKNN Classifier on RTS 79. 28

Figure 3.2: Flowchart of Overall Process of MCS-MLKNN Method 31

Figure 3.3: Overall Diagram of Proposed Classifier for Variable Load Levels. 40

Figure 4.1: Flowchart Describing Training Phase of MLRBF 53

Figure 4.2: Flow Chart of Importance Sampling. 57

Figure 4.3: Overall Diagram of Proposed Method 59

Figure 4.4: Overall Diagram of Proposed Classifier for Variable Load Level 64

Figure 5.1: Proposed Algorithm used to create a Proportional Training Dataset 76

Figure 5.2: Overall Diagram of Proposed CNN Structure. 79

Figure 5.3: General Structure of Proposed Method 81

LIST OF TABLES

Page

Table 3.1: Comparison of Overall System Performance for IEEE 30 Bus Test System .35

Table 3.2: Comparison in Bus Level for IEEE 30 Bus Test System...............................36

Table 3.3: Comparison of Overall System Performance for IEEE RTS..........................37

Table 3.4: Comparison of MLKNN & CMCS in Bus Level 38

Table 3.5: Comparison of System Performance for Varying Hourly Load 41

Table 3.6: Comparison of System Performance for Varying Load in Bus Level 41

Table 3.7: Performance Comparison when Transmission Line Failures Considered 44

Table 3.8: Comparison of Bus Level Classification Performance for Case 4..................44

Table 3.9: Performance for System Failures Sourced from Transmission Line Failures 45

Table 4.1: Comparison on Overall Performance Analysis and CPU Time Spend...........62

Table 4.2: Comparation of Classification Performance at Bus Level Based on LOLP ...62

Table 4.3: Comparison on Overall Performance Analysis for Varying Load Levels 64

Table 4.4: Comparison of Performance at Bus Level for Varying Load Levels.............65

Table 5.1: Performance Comparison of CNN-MCS & CMCS for Constant Load (DC).83

Table 5.2: Performance Comparison of CNN-MCS & CMCS for Constant Load (AC).83

Table 5.3: Performance Comparison of CNN-MCS & CMCS in Varying Load (DC) ...85

Table 5.4: Performance Comparison of CNN-MCS & CMCS in Varying Load (AC) ...86

CHAPTER I

INTRODUCTION

This chapter first presents the background of the research described in this dissertation and later research objectives and the structure of dissertation are given explaining an overall picture of this study.

A. Introduction

Tremendous amount of research has been done on developing probabilistic methods for power system reliability evaluation over the past several decades. In most cases, those methods can be group in two categories as analytical solution methods or simulation-based methods. In analytical methods the system is represented by mathematical models and reliability indices are computed using mathematical solutions. Even though those methods give exact solutions within the assumptions made, deriving those models can become a challenging problem especially for large power systems. Among the many simulation methods developed, including Importance Sampling (IS) and Latin Hypercube Sampling (LHS), Monte Carlo Simulation (MCS) based techniques are currently the most widely used methods to estimate the reliability indices of large composite power systems. The reason for this is the flexibility that these methods provide in incorporating some of the system complexities. MCS methods use, random sampling of system states with the basic concept that their sampling is proportional to their probabilities of occurrence. For most cases, MCS is more suitable for composite system analysis because of its simplicity and flexibility in estimating complex system parameters in various conditions [1, 2]. Despite the advantages of MCS, just like analytical methods it requires solving optimization equations to perform optimal power flow (OPF) analysis for characterization of each sampled state and repetitive states most of the time. Therefore, MCS suffers from long computation time to produce statistically converged reliability indices especially for high reliability systems. This poses a need for research for efficient simulation methods in reliability analysis of large power systems. Considerable amount of research has been done on increasing computational efficiency of these simulations in the past. Some of these approaches use variance reduction techniques [3], state space pruning [4], fuzzy optimal power flow [5] or more efficient sampling techniques like LHS [1] or IS [6]. Some of these researches also use population-based intelligent search (PIS) methods as an alternative to search for meaningful states to decrease the computational burden of these simulation methods. Some of classical examples of these methods are genetic algorithms (GA) [7,8], particle swarm optimization (PSO) [9] or ant colony optimization (ACO) [10]. Some of the researchers also implement pattern classification techniques to reduce the number of states to be evaluated in power system reliability assessment. Some examples of these methods are Artificial Neural Network (ANN) based classifiers [11], Artificial Immune Recognition System (AIR) [12] or Least Squares Support Vector Machine based classifiers [13]. Pattern classification-based techniques have shown significant performance to reduce the computational burden required for reliability analysis however there is still more research required for increasing classification accuracy and model flexibility. In this dissertation novel methodologies to perform composite system reliability evaluation through Artificial Intelligence based MCS is investigated to increase computational efficiency and classification accuracy of simulation. Research has been done for this purpose that explores deep learning and multi-label classification methods. In addition to that combination of multi-label classification and Importance Sampling is explored as a separate chapter.

B. Research Objectives

The broad focus of this dissertation is the development of novel AI based computation methods that increase the performance of MCS applied to the evaluation of composite power system reliability. This increase in performance is in computational efficiency, computational accuracy and including factors not done previously like computation of bus indices. The contributions of this dissertation are listed below:

1. The design and development of different types of deep learning structures for MCS based composite system reliability evaluation.

2. Investigating the performance of multi-label learning in MCS based composite system evaluation.

3. Demonstrate the performance of combination of importance sampling and multi-label learning for composite system reliability evaluation.

C. Dissertation Structure

As stated in the previous section, this dissertation focuses on improving the performance of the evaluation of composite power system reliability through the development of novel algorithms using deep learning and multi-label learning approaches. In the second chapter, background information is provided about composite system adequacy analysis and MCS techniques. In the third chapter, different types of deep learning structures are investigated and performance of those algorithms demonstrated on case studies. Chapter IV first gives background information about multi-label learning and then performance of this type of learning for composite system reliability evaluation is explored through case studies. Chapter V focuses on combination of multi-label learning and importance sampling combination within MCS. After briefly explaining the theory behind importance sampling, performance results for the proposed method are presented in this chapter. Finally, the research then concludes in Chapter VI.

CHAPTER II

CONCEPT OF COMPOSITE POWER SYSTEM RELIABILITY EVALUATION AND MONTE CARLO SIMULATION

There are two main categories of power system reliability evaluation techniques, analytical and simulation based. In analytical modeling, a model is built that reasonably approximates the physical system and is also amenable to calculations. Monte Carlo Simulation methods, on the other hand, are based on sampling and estimating the indices from the samples generated. MCS based techniques are able to handle any type of probability distribution associated with component state durations, capture systematic and temporal dependencies, and evaluate probability distributions of resultant indices. In general, they provide more flexibility to incorporate complex operating conditions in assessing especially large and complex power systems compared to analytical solutions [14,15]. In this section first MCS is introduced and later power system models, test systems and metrics used to measure performance of AI based power system reliability evaluation methods are described.

A. Monte Carlo Simulation

Monte Carlo simulation is a representative simulation method that is usually adopted to deal with reliability evaluation of large-scale or complex power systems. MCS methods are classified into two categories, non-sequential simulation and sequential simulation. Nonsequential simulation is based on a random sampling algorithm, where a component state is selected according to its probability distribution without considering chronological connections. By using this approach, reliability indices such as loss of load probability (LOLP), and expected unserved energy (EUE) can be directly obtained but loss of load frequency (LOLF) and loss of load duration (LOLD) may need additional assumptions.

However, previous researches show that these indices can be estimated using a frequency balance concept with negligible increase in computational burden to overcome this difficulty. Calculation of these indices can be estimated through a conditional probability approach [16,17], or calculated directly from sampled failure states using the frequency balance property [18]. Main handicap of non-sequential approach emerges when a system chronology is required to reflect the inherent variability of reliability estimations or to incorporate time-varying characteristics.

Sequential Monte Carlo Simulation approach becomes more suitable [19] in these situations. As an example, if aging factor is considered as a practical issue in reliability, then component failure rates that increase with time are naturally incorporated by sequential simulation [20]. Sequential simulation steps through system states in time domain, where a state of each component is chronologically connected to its adjacent states. A realistic history is created by combining sequences of component state durations and system load over a time horizon. In this manner, sequential simulation estimates LOLP, EUE, LOLF or LOLD indices more accurately. Also, economic indices such as loss of load cost (LOLC) can be estimated accurately [21]. Compared to non-sequential approach, sequential simulation provides simplicity of accurately incorporating time-dependent variables and their correlations, however, this approach requires considerably more computing time to converge than non-sequential simulation. In non-sequential simulation, any two sampled system states can be completely independent, on the other hand, in sequential simulation, any two consecutive system states differ by a realization of one random variable. As a result, the overall state space is less represented by sequential simulation than by non-sequential simulation considering the same number of sampled states. Therefore, sequential simulation would require a larger number of states to reach the same convergence criterion. This problem is especially critical for composite systems where their state evaluation involves analysis of power flow and optimization-based remedial action.

1. Non-Sequential Monte Carlo Simulation

In non-sequential MCS approach system states are sampled from the state space using the concept of proportionality, i.e., states are sampled proportional to their probability. Followingly non-sequential MCS is described in three main steps.

1.) Select a state of the power system by sampling the states of all components and the load levels.

2.) Characterize the selected state as success or failure through test function, by performing the adequacy analysis, which usually involves optimal power flow (OPF) analysis.

3.) Update the estimate $E(F)$, the expected value of the system reliability indices using the results obtained in step 2. $E(F)$ is described in (eq 2.1).

$$E(F) = \frac{1}{N}\sum_{i=1}^{N} F(x_i) \tag{2.1}$$

where N is the number of simulated states.

4.) If the stopping criterion is satisfied then stop the simulation, otherwise, return to step 1.

The estimate of uncertainty is usually represented by the coefficient of variation β. An acceptable value of the estimate of uncertainty is used as a stopping criterion for the simulation. Besides variance, pre-specified number of samples can also be used as stopping criteria. Calculation of β is described as (eq 2.2) below.

$$\beta = \frac{\sqrt{V\big(E(F)\big)}}{\big(E(F)\big)} \tag{2.2}$$

where $V\big(E(F)\big)$ is the variance of the estimate $E(F)$.

2. Sequential Monte Carlo Simulation

In sequential MCS each subsequent system state is related to the previous set of system states. By doing this a sequential time evaluation of system behavior is created which enables evaluation of a wide range of reliability indices [22]. Sequential simulation can be generally implemented with two methods, fixed time interval method and next event method. Both methods are described followingly.

a. Fixed Time Interval Method

In this method, sequence of time intervals is stepped through, where component states are selected according to their transition probabilities. Its steps are described as follows.

1.) Initialize component states with random sampling from their probabilities of being up or down.

2.) Sample for component states in next transition using each component transition probability matrix in (eq 2.3), where $\Delta\tau$ is a chosen as small time step.

$$\begin{array}{cc} & \begin{array}{cc} Up & Down \end{array} \\ \begin{array}{c} Up \\ Down \end{array} & \begin{bmatrix} 1-\lambda\Delta\tau & \lambda\Delta\tau \\ \mu\Delta\tau & 1-\mu\Delta\tau \end{bmatrix} \end{array} \tag{2.3}$$

3.) Generate a load level for step $\Delta\tau$ from historical chronology.

4.) Evaluate current system state with contingency analysis. If no bus has loss of load then load curtailment is zero otherwise, remedial action is called to find a load curtailment.

5.) Repeat Steps 2–4 while updating reliability indices. If convergence criterion is satisfied, stop the process.

The length of time step $\Delta\tau$ will affect simulation accuracy. A smaller step results in higher accuracy, but will require a larger number of states to be evaluated and thus incur higher computational cost. This issue imposes a computational limitation for fixed time interval method to be applied in practice even though it is theoretically feasible.

b. Next Event Method

In this method, simulation proceeds by keeping a record of the time when the next event occurs, where the residence time of each component state is determined by the value of a random variable from its continuous distribution. Its steps are given as follows:

1.) Initialize component states with random sampling from their probabilities of being up or down.

2.) Generate the state (up or down) duration τ for each component i. Draw a pseudo-random number $z \sim U(0,1)$ and substitute it into the inverse transform of distribution function F_t in (eq 2.4).

$$\tau_i = F_{t_i}^{-1}(z) \tag{2.4}$$

3.) Update the associated load sequence in correspondence to component sequence.

4.) Evaluate each state of system sequence obtained in Steps 2–3 in the similar way as seen in Step 4 of Fixed time interval method.

5.) Repeat Steps 2–4 while updating reliability indices. If convergence criterion is satisfied, stop the procedure.

B. Composite System Adequacy Analysis

MCS techniques are currently the most widely used methods to assess the adequacy analysis of a composite system [23]. The MCS is based on a combination of state sampling with direct approach for system analysis and the utilization of a minimization model for load curtailment. This method is especially well suited for large power systems and allows multi state representation of components as well. Minimization model of load curtailment usually requires to solve an optimization problem based on power flow equations. The power flow equations used for analysis usually use either DC or AC power flow model. Following subsections present the mathematical information required to perform load curtailment analysis based on DC or AC power follow model.

1. DC Power Model

In composite system reliability studies, power flow analyses are usually carried out in solving optimization problems for minimum load curtailment. There are usually two types of power flow analysis used to characterize a system state, DC and AC flow analysis. The DC power flow model is described by the nodal equation

$$\beta\delta + G = D \tag{2.5}$$

and the line flow equation

$$bA\delta = F \tag{2.6}$$

where

$\mathrm{N_b}$ = Number of Buses

$\mathrm{N_t}$ = Number of Transmission Lines

b = $\mathrm{N_t x N_t}$ primitive matrix of transmission linesusceptances

A = $\mathrm{N_t x N_b}$ element node incidence matrix

β = $\mathrm{N_b x N_b}$ augmented node susceptance matrix

δ = $\mathrm{N_b}$ vector of bus voltage angles

G = $\mathrm{N_b}$ vector of bus generation levels

D = $\mathrm{N_b}$ vector of bus loads

F = $\mathrm{N_t}$ vector of transmission line flows

Load curtailment can be found by solving following linear programming model

$$\text{Loss of Load} = \min\left(\sum_{i=1}^{N_b} C_i\right) \tag{2.7}$$

Subject to

$$\beta\delta + G + C = D$$

$$G \leq G^{max}$$

$$C \leq D$$

$$bA\delta \leq F^{max} \tag{2.8}$$

$$-bA\delta \leq F^{max}$$

$$G, C \geq 0$$

$$\delta, unrestricted$$

2. AC Power Flow Model

Following set of equations describe the formulation and incorporation of the objective function of minimum load curtailment in the non-linear programming problem. This objective function is subject to equality and inequality constraints of the power system operation limits. The equality constraints include the power balance at each bus and the inequality constraints are the capacity limits of generating units, power carrying capabilities of transmission lines, voltage limits at the nodes and reactive power capability limits. The minimization problem is formulated as follows [24]

$$\text{Loss of Load} = \min\left(\sum_{i=1}^{N_b} C_i\right) \tag{2.9}$$

Subject to

$$P(V,\delta) - P_D + C = 0$$

$$Q(V,\delta) - Q_d + C_q = 0$$

$$P_G^{min} \leq P(V,\delta) \leq P_G^{max}$$

$$Q_G^{min} \leq Q(V,\delta) \leq Q_G^{max} \tag{2.10}$$

$$V^{min} \leq V \leq V^{max}$$

$$S(V, \delta) \leq S^{max}$$

$$0 \leq C \leq P_D$$

$$\delta, unrestricted$$

In (1) and (2), C_i is the load curtailment at bus i, C is the vector of load curtailments, C_q is the vector of reactive load curtailments, V is the vector of bus voltage magnitudes, δ is the vector of bus voltage angles, P_D and Q_D are the vectors of real and reactive power loads, P_G^{min}, P_G^{max}, Q_G^{min} andQ_G^{max} are the vectors of real and reactive power limits of the generators, V^{min} and V^{max} are the vectors of minimum and maximum allowed voltage magnitudes, S (V, δ) is the vector of power flows in the lines, S^{max} is the vector of power rating limits of the transmission lines and P(V, δ) and Q(V, δ) are the vectors of real and reactive power injections. Moreover, N_b is the number of buses, $N_d$is the number of load buses, N_t is the number of transmission lines and N_g is the number of generators. In the standard minimization problem given by (1) and (2), all generation and network constraints have been taken into consideration. It has been assumed that one of the bus angles is zero in the constraints (2) to work as a reference bus.

C. Reliability Test Systems

IEEE 30 bus test system or IEEE RTS is used to demonstrate performance of the proposed methods in this study. In this subsection, test systems are described followingly.

1. IEEE 30 Bus Test System

There are 41 transmission lines in this system, with 435 MW maximum generation and 255 MW maximum load. There are 9 generation units for 6 generation buses in this case study. Since there is no available reliability data associated with this system and therefore corresponding parameters are chosen from RTS. For simplicity, all generators are assumed to share the same failure rates and repair time and all transmission lines share same failure rates and repair time. A detailed schematic of IEEE 30 Bus Test System is given in figure 1. Data of the original IEEE 30 Bus Test System can be accessed through the [25].

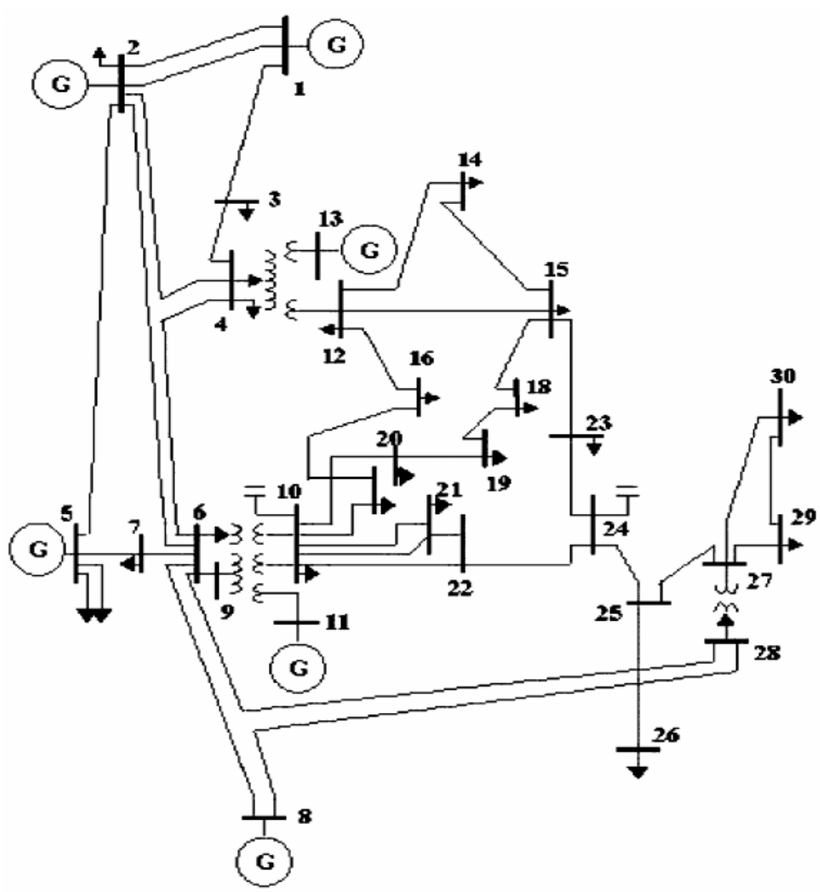

Figure 2.1: Schematic for IEEE 30 Bus Test System

2. IEEE RTS 79 Test System

The system has 24 buses (10 of them are generation buses), 38 transmission lines and 32 generation units. The total installed capacity is 3405 MW and the system has 2850 MW at its annual peak. A detailed schematic of RTS is given in figure 2. Data of the original IEEE RTS can be accessed through the [26].

For some of case studies Modified RTS (MRTS) is preferred. MRTS is designed for the studies on transmission line reliability studies for composite systems. In this system, generation capacities are doubled and all the loads are multiplied by 1.8 while rest of the system parameters remain unchanged. In this way effect of transmission lines on overall system reliability is increased and become more observable. The total installed capacity is 6810 MW and the system has 5130 MW at annual peak.

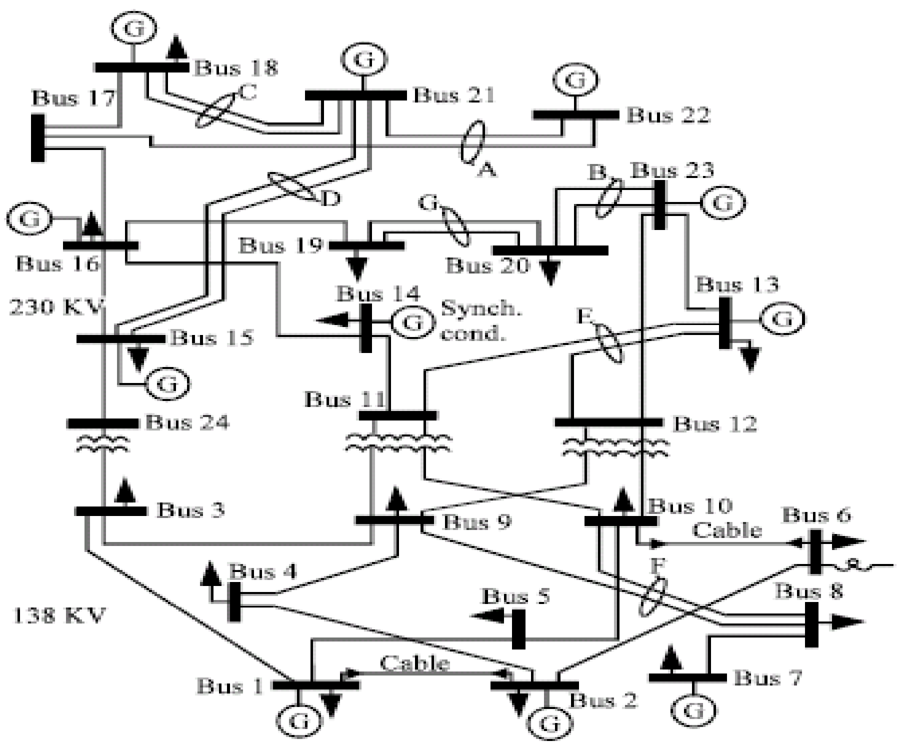

Figure 2.2: Schematic for IEEE RTS

D. Performance Evaluation Metrics

Performance of an AI based composite system reliability evaluation techniques is commonly considered as binary classification. To test performance of those systems usually statistical measures of Sensitivity (also termed as recall) and Specificity are utilized. These measurements are based on the terminologies of True Positive (TP), False Positive (FP), True Negative (TN), and False Negative (FN). Here, True and False refers the assigned classification being correct or incorrect, while positive or negative refers to assignment to the positive or the negative category.

In a binary classification, sensitivity expresses proportion of correctly identified positives to all predictions classified as positive. Calculation of sensitivity is described in (eq 2.11).

$$Sensitivity = \frac{TP}{TP+FN} \tag{2.11}$$

Sensitivity measures the performance of a classifier in reducing computational time of MCS since each False Negative requires an analysis by power flow equation. Specificity on the other hand, measures the proportion of correctly identified negative samples of all samples that are classified as negative. Specificity is described in (2.12).

$$Specificity = \frac{TN}{TN+FP} \tag{2.12}$$

In terms of reliability evaluation, specificity measures the accuracy of the classifier in estimating reliability parameters since incorrect classifications of negative cases tend to change calculated reliability parameters. In addition to the metrics described above, performance of a proposed method in estimating composite power system reliability indices evaluated based on Loss of Load Probability (LOLP). Definition of LOLP is given in (2.13).

$$LOLP = \frac{Total\ number\ of\ Failures}{TotalNumberof\ Samples} \tag{2.13}$$

CHAPTER III

MULTI-LABEL CLASSIFICATION FOR COMPOSITE POWER SYSTEM RELIABILITY EVALUATION[1]

In this chapter a new approach for reliability evaluation of composite power systems by combining Monte Carlo simulation (MCS) and Multi-Label Learning (MLL) is described. Multi-Label K-Nearest Neighbor (MLKNN) algorithm is used as classifier to show effectiveness of the proposed method. MLL is a classification technique in which target vector of each instance is assigned into multiple classes. In this research MLL methods is used to classify states (failure or success at bus level) of a complete power system without requiring optimal power flow (OPF) analysis, except in the training phase. As a result, the computational burden to perform OPF is reduced dramatically. For illustration, the proposed method is applied to the IEEE 30 BUS Test System and IEEE Reliability Test System (IEEE RTS). The obtained results from various case studies demonstrate that MLKNN based reliability evaluation provides promising results in both classification accuracy and computation time in evaluating the composite power system reliability. Details of the proposed method are presented in following subsections.

---

1 Part of this section is reprinted from copyrighted material with permission from IEEE. © 2018 IEEE. Reprinted, with permission, from Dogan Urgun and Chanan Singh, "A Hybrid Monte Carlo Simulation and Multi-Label Classification Method for Composite System Reliability Evaluation", *IEEE Transactions on Power Systems*, November 2018.

A. Multi-Label Learning for Power System Reliability Evaluation

Multi-label classification is a type of learning where each sample is associated with multiple labels, making it suitable for calculation of bus indices. The multi-label learning methods can be explored in two main groups which are algorithm adaptation and problem transformation methods. Algorithm adaptation methods mainly target to extend some specific single class learning algorithms to handle multi-label classification problems directly. Some examples of this group include MLKNN, neural networks based Multi-Label classification or decision trees. The transformation methods, on the other hand, aim to transform a multi-label classification problem into a single label classification problem. Binary reverse method or pair-wise method can be given as examples for this method. In this part of research, a combination of MCS and MLKNN classifier is used to evaluate reliability indices of composite power systems. The main contribution of proposed method is presenting a method that minimizes computational burden of classification of sampled states with MCS and reduces the requirement for OPF for the reliability evaluation, except in the training stage and to extend the capability to bus level classification. MLKNN has one of the most time efficient structure among many MLL methods. This feature allows increasing computational efficiency of MCS. Moreover, experiments show that performance of MLKNN is superior to those of some well-established multi-label learning methods [27,28,29,30,31].

B. Multi-Label K-Nearest Neighbor Algorithm

MLKNN approach is an MLL algorithm which is derived from the traditional KNN. In this method, KNN for each element in the training set is identified. Then statistical information is gained from the label sets for each instance. Lastly maximum a posteriori (MAP) principle is applied to determine the label for the test instance. In this study MLKNN algorithm is chosen for evaluation of reliability indices of composite power system because of both classification performance and time efficient structure.

Before explaining the algorithm, several notations are introduced. Let there be an instance m and its associated label set $Y \subseteq \acute{y}$. Let $\overrightarrow{y_m}$ be the category vector for m, where its q_{th} component $\overrightarrow{y_m}(q)$ $(q \in Y)$ takes the value of 1 if $q \in Y$ and 0 otherwise. In addition, let N (m) denote the set of KNNs of m identified in the training set. Thus, based on the label sets of these neighbors, a membership counting vector can be defined as:

$$\overrightarrow{C_m}(q) = \sum_{a \in N(m)} \overrightarrow{y_a}(q), \; q \in y \tag{3.1}$$

Where $\overrightarrow{C_m}(q)$ counts the number of neighbors of m belonging to the qth class. This vector is used to determine how many samples in number of neighbors N of sample m has labeled for each class described. In terms of composite system evaluation, the equation describes how many load failures occurred for the sample m in K number of neighbors. These numbers can be obtained by counting the occurrence of failures in training target matrix.

For each test instance t, MLKNN firstly identifies its KNNs N (t) n the training set. Let H_1^q be the event that t has label q, while H_0^q be the event that t does not have label l. Furthermore, let E_j^q ($j \in \{0,1, \ldots, K\}$) denote the event that, among the KNNs of t, there are exactly j instances which have label q. Therefore, based on the membership counting vector $\overrightarrow{C_t}$ the category vector $\overrightarrow{y_t}$ is determined using the following maximum a posteriori principle:

$$\overrightarrow{y_t}(q) = \operatorname{argmax}_{b\in\{0,1\}} P\left(H_b^q | E^q_{\overrightarrow{C_t}(q)}\right),\ l \in Y \tag{3.2}$$

Using the Bayesian rule, Eq. (2) can be rewritten as:

$$\overrightarrow{y_t}(q) = \operatorname{argmax}_{b\in\{0,1\}} \frac{P(H_b^q)P\left(E^q_{\overrightarrow{C_t}(q)}|H_b^q\right)}{P\left(E^q_{\overrightarrow{C_t}(q)}\right)} \tag{3.3}$$

$$= \operatorname{argmax}_{b\in\{0,1\}} P(H_b^q) P\left(E^q_{\overrightarrow{C_t}(q)}|H_b^q\right) \tag{3.4}$$

$$P(q) = P\left(H_1^q | E^q_{\overrightarrow{C_t}(q)}\right) = \frac{P(H_1^q)P\left(E^q_{\overrightarrow{C_t}(q)}|H_1^q\right)}{\sum_{b\in\{0,1\}} P(H_b^q)P\left(E^q_{\overrightarrow{C_t}(q)}|H_b^q\right)} \tag{3.5}$$

Eq. (3.2 – 3.4) explain how to calculate prior and conditional probabilities. Prior probability term is used to describe the loss of load probability for each bus in overall training dataset. The output of this process is a $\mathrm{Qx1}$ vector where Q is number of buses in a system. Conditional probability terms represent the loss of load probabilities for each bus of sample m in all K neighbors. This probability is also calculated by counting the occurrences of failures in all K neighbors for each bus Q. The output of this process is a $\mathrm{Qx}(\mathrm{K}+1)$ matrix where Q is number of buses in a system and K is the number of neighbors specified for classifier.

As shown in Eq. (3.4), in order to determine the category vector $\overrightarrow{\mathrm{y_t}}$, all the information that is needed is the prior probabilities $\mathrm{P}\left(\mathrm{H_b^q}\right)$ $(\mathrm{j} \in \{0,1, \dots, \mathrm{K}\})$. Actually, these prior and posterior probabilities can all be directly estimated from the training set based on frequency counting.

Correspondingly, $\mathrm{C}'[\mathrm{j}]$ counts the number of training instances without label q whose k nearest neighbors contain exactly j instances with label q. Finally, using the Bayesian rule, steps from (3.5) the algorithm's outputs based on the estimated probabilities can be computed.

C. MLKNN for Power System Reliability Evaluation

In this section, first formulation of general composite system reliability evaluation parameters is made, later, implementation of MLKNN algorithm is fully explained by steps.

1. General Definition of MLKNN Parameters

In this study, total generation capacities for buses of composite system are taken as input parameter for MLKNN classifier. So, each bus which is capable of generation in the system is considered as an element of input matrix G for every sample (instance) M as described in (3.6).

$$G_{input} = \begin{bmatrix} G_{11} & G_{12} & G_{1N} \\ G_{21} & G_{22} & G_{2N} \\ G_{M1} & G_{M2} & G_{MN} \end{bmatrix} \tag{3.6}$$

Where N is the number of the generation buses and M is the total number of samples in the input matrix.

A target matrix T is also created for training of the MLKNN classifier which includes state information for each bus of the system for M different samples, described in (3.7).

$$T = \begin{bmatrix} T_{11} & T_{12} & T_{1Q} \\ T_{21} & T_{22} & T_{2Q} \\ T_{M1} & T_{M2} & T_{MQ} \end{bmatrix} \tag{3.7}$$

Where Q is the number of the load buses in the system and S is the status information of bus q. While defining status of buses '-1' is taken to describe 'success state' and '1' for 'failure state'.

Desired output for this classifier P_{out}, contains failure probabilities for each bus of composite reliability system for each sample M, described in (3.8).

$$P_{out} = \begin{bmatrix} P_{11} & P_{12} & P_{1Q} \\ P_{21} & P_{22} & P_{2Q} \\ P_{M1} & P_{M2} & P_{MQ} \end{bmatrix} \tag{3.8}$$

2. Explanation of MLKNNN Procedure

After giving definitions of general parameters for MLKNN classifier, the training and testing procedure is now explained in steps. Before starting explanation, a few parameters are described:

m: defines index of current sample of the total M samples.

q: defines the bus index of total Q buses of system.

T_m defines the state of bus *q* at sample *m* so;

$$T_m(q) = \begin{cases} -1 \; where\ bus\ q = success \\ \;\; 1 \;\; where\ bus\ q = failure \end{cases}$$

K indicates the determined nearest neighbor index used in classification.

Training:

1. MLKNN is a classification technique which uses the k-nearest neighbor algorithm for finding closest relation between training samples. So, the first step of training procedure is creating a distance matrix. In this study, Euclidian distance method is used to create this matrix described in (3.9).

$$\sum_{i=1}^{M} \sqrt{(a_i - b_i)^2} \tag{3.9}$$

For further explanation, a vector is described to represent sum of squares for input vectors for each sample in (3.10).

$$G_{ss} = [G_1^2 + G_2^2 \dots G_N^2] \tag{3.10}$$

Where G_{ss} sum of squares for each generation bus and N is the number of total generation buses. Based on equation (3.10) a concurrent generation matrix can be described in (3.11).

$$G_{concur} = \begin{bmatrix} G_{ss1} & G_{ss1} & G_{ss1} \\ G_{ss2} & G_{ss2} & G_{ss2} \\ G_{ssM} & G_{ssM} & G_{ssM} \end{bmatrix} \tag{3.11}$$

Where G_{concur} is the concurrent matrix created to calculate the distance used in k-mean algorithm and M is number of total samples. At this point G_{concur} is applied to equation (9) described at (3.12).

$$Dist = \sqrt{G_{concur} + G_{concur}^T - 2\left(G_{input} x G_{input}^T\right)} \tag{3.12}$$

Where Dist is the matrix including the data of distances between samples. Finally, the distance matrix is described in (3.13) below.

$$Dist = \begin{bmatrix} Dist_{11} & Dist_{12} & Dist_{1M} \\ Dist_{21} & Dist_{22} & Dist_{2M} \\ Dist_{M1} & Dist_{M2} & Dist_{MM} \end{bmatrix} \tag{3.13}$$

Where each element of the MxM matrix describes the distances between samples.

1- In the second step prior probabilities of failure for each bus are calculated based on counting instances as shown in (3.1). Calculation of prior probabilities is described in (3.14).

$$P^1(q) = \frac{\sum_{i=1}^{M} Y_i(q)}{M} \tag{3.14}$$

$$P^0(q) = 1 - P^1(q) \tag{3.15}$$

At the end of process, a Qx1 Prior and a Qx1 Compliment Prior probability matrix are obtained which gives prior probabilities of each bus.

2- In the third step conditional and conditional compliment probabilities for buses are calculated for K nearest neighbors based on counting. In this step the algorithm first determines how many of K nearest neighbors for sample m have failure at bus q. Later the process is repeated for all M samples to determine probability of failure for bus q conditional to the occurrence in nearest numbers. This process is formulated in (3.16).

$$Cond(k, q_b) = P\left(C^q_{(k)\in\{0,K\}]} | H^q_{b\in\{0,1\}}\right) \tag{3.16}$$

where $C^q_{(k)\in\{0,K\}]}$ denote the number of instances which have failure on bus q. Also H^q_1 describes the event that bus q has failure at the sample x as likewise H^q_0 describes the event bus q does not have failure at the sample x.

At the end of these steps four required probability matrices for the system are obtained.

-Prior (Q) is a Q x 1 matrix that defines prior probability of failure for all buses in system.

-Prior Negative (Q) is a Q x 1 matrix that defines prior probability of failure for all buses in system.

-Cond ($K|Q$) is a Q x (K+1) matrix that shows conditional probabilities of failure for all buses in system according to kth closest neighbor.

-Cond Negative ($K|Q$) is a Q x (K+1) matrix that shows negative of conditional probabilities of failure for all buses in system according to kth closest neighbor.

After training parameters are obtained, testing process is used to calculate probability of failure for each bus for given sample m based on using Bayesian rule showed in (3.5).

Testing:

After training of MLKNN classifier is completed, testing process can be used to identify bus statuses of a composite power system. Testing process has 3 main steps:

1- As in training, the first step of testing is also calculating distance matrix between test data and training data. The same process as described in (3.9) is used in this step. At the end of this step M_{test} x M_{train} distance matrix is obtained.

2- In this step, the number of failures in K nearest neighbors is determined based on the counting process (3.1). Results of the counting indicates the required indices for Cond $(k|q)$ where q indicates the bus number and $k \in \{0, \ldots, K\}$.

3- In the last step Bayesian rule described in (3.5) is used for determining failure probabilities of busses in test database.

$$P(q) = \frac{P(q)P(k|q)}{\sum_{b\in\{0,1\}} P\left(C_{(k)}^{q}|H_{b}^{q}\right)} \tag{3.17}$$

After the probability matrix is obtained, previously specified threshold can be used to determine if a bus is in failure state or not. Overall flow diagram for the proposed MLKNN classifier for IEEE RTS 79 test system which has 24 buses (10 of them are generation buses), 38 transmission lines and 32 generation units. is given in figure 3.1.

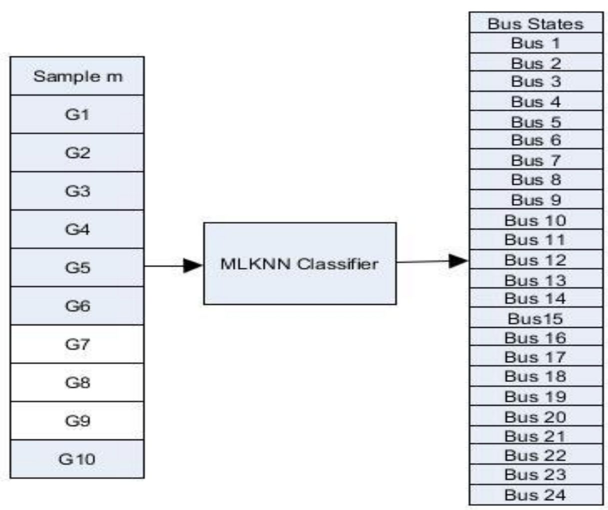

Figure 3.1: Overall Diagram for MLKNN Classifier on RTS 79

D. Proposed Topology

This section explores the MLKNN classifier in conjunction with the MCS to reduce the computational requirements while evaluating the composite power system reliability. In this section firstly, the basic steps of the MCS are described and then detailed computational procedure of the proposed MLKNN classifier for the composite power system reliability evaluation is explained.

There are two types of MCS approaches commonly used in composite system evaluation, non-sequential and the sequential. The basic difference between these approaches is that the samples are determined using proportionality concept in non-sequential approach while sequential approach, identifies system states by considering the chronological characteristics of the system. In this paper, the non-sequential MCS is used as a benchmark for the testing the performance of the proposed method. Non-sequential MCS is generally preferred for composite test system because of simplicity of method and its computational efficiency. The basic steps for the composite reliability evaluation by the non-sequential MCS are explained as follows;

1- Select a random state for all components of the power system as $x = (x_1, x_2 \dots x_m)$ where m is the component number in power system.

2- Classify each of the selected state x (as success or failure) through KLMNN classifier (classifier is trained with a proper training database created in section A).

3- Update the estimate E (F), the expected value of the system reliability indices using the results gathered from step 2 described in (3.18).

$$E(F) = \left(\frac{1}{N}\sum_{i}^{N} x_i\right) \tag{3.18}$$

Where N is the number of simulation steps.

4- If the determined criteria for variance is reached stop the simulation otherwise return to the first step.

The estimate of uncertainty is usually represented by the coefficient of variation β. An acceptable value of the estimate of uncertainty is used as stopping criteria for the simulation. Besides variance, determined specified number of samples can also be used as stopping criteria.

Composite power systems are generally highly reliable systems, so the probability that failed state occurs is much less than success states. Therefore, the steps described in this section repeat many times. As a result, reducing the computational burden of power flow analysis by using KLMNN classifier provides significant time and computational efficiency.

The first step of implementing MLKNN classifier in composite system evaluation with combination of MCS is generating a training database. A proper dataset is created using a set of sampled states and the corresponding state classification labels for each bus (success or failure), which are obtained from the MCS. Once the appropriate training patterns are obtained, then the MLKNN classifier is trained, which would then be used for the state space classification of the testing database to evaluate the reliability. In this database, input vector is created by using generation buses while output of the classifier is defined as state (success or failure) of each bus of the selected test system. Proposed method is applied on two different test systems; IEEE 30 Bus Test System (case 1) and IEEE RTS 79 (case 2). Performance of proposed system is also tested on varying load levels (case 3) and under circumstance of failing transmission lines (case 4). In these test systems input data sampled from 6, 10, 24 and 47 input buses respectively while target matrices sampled from 24 and 30 buses for RTS and IEEE 30 Test System. General algorithm of the proposed method for composite reliability evaluation is shown in Fig. 3.2 and its detailed implementation procedure is outlined below.

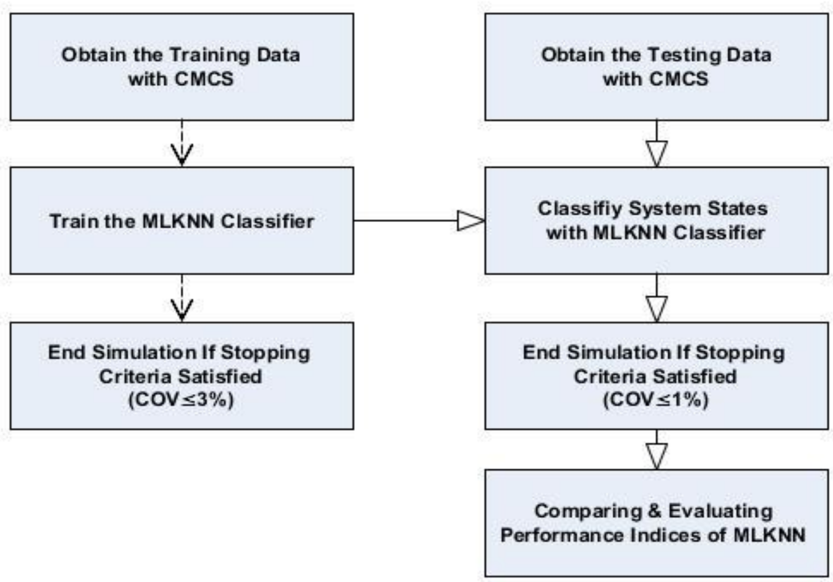

Figure 3.2: Flowchart of Overall Process of MCS-MLKNN Method

Training data samples for MLKNN classifier are obtained by the MCS. Size of these data samples can be determined by either using pre-specified number of samples or the convergence of the coefficient of variation method. In this study, size of dataset determined by specific number of samples proportional to the LOLP of the system to get the sufficient number of attributes in the input training patterns. In this training datasets, input vectors are created based on information of total generation for each bus, and target matrices are created by using corresponding system state characteristics (success or failure) for each bus of the testing system. In the target matrix 1 and -1 are selected to represent the desired output of the success and failure states, respectively. In this study, the repetitive states of MCS are not included in the training database to reduce the number of samples included to the training process. It should be also noted that the number of obtained success states is much higher than failed states, some of the success states are discarded to prevent overemphasis of classification. Consequently, the number of training patterns is decreased to speed up the MLKNN training and a balanced training patterns is created to increase overall performance of classifier.

After training patterns have been generated, the next step is training the MLKNN for selected set of input/output patterns. Once the MLKNN classifier is trained, the MCS follows the same steps as described early in this section with the exception that the state characterization is now performed by the trained MLKNN instead of running the DC-OPF calculations. With this procedure, the composite reliability indices can be calculated without requiring power flow calculations. In this way, computational time necessary for evaluating the composite power system reliability is reduced dramatically. In this study, the simulations run until coefficient of variation reaches (COV) 1% for testing stage in all case studies. COV indices are calculated based on overall failures of specified test system. That is because overall the failure rate of a system is much higher than bus level failures. The overall performance of the MLKNN classifier is measured by using the parameters: overall accuracy, sensitivity, specificity, g-mean and simulation time.

E. Case Studies and Results

In this section four case studies were conducted on IEEE 30 Bus System and IEEE RTS to analyze the performance of MLKNN classifier.

In case study one, the performance of the proposed method is demonstrated on IEEE 30 Bus System. Following case studies are performed on the IEEE RTS since it is used by most of the developers of new algorithms for composite system reliability evaluation studies. Performance of the proposed method on RTS shown in case study two and case study three are for peak load level and hourly varying load level respectively. For the first three case studies the capacity and admittance constraints of transmission lines are, considered. However, the transmission line failures are not considered as these have much smaller probabilities than the generator failures.

In case study four, transmission line failures are also considered. For this case study, Modified RTS (MRTS) is preferred. MRTS is designed for the studies on transmission line reliability studies for composite systems. In this system, generation capacities are doubled and all the loads are multiplied by 1.8 while rest of the system parameters remain unchanged. In this way effect of transmission lines on overall system reliability is increased and become more observable.

All the simulations are performed using MATLAB (2017b) platform on a PC with Intel Core i7-4510 CPU (~2.6GHz), 16 GB Memory. Simulation results for case studies are discussed in following subsections.

1. Performance on IEEE 30 Bus Test System

In case study 1, to test performance of the proposed classifier, the system is tested on IEEE 30 Bus Test System for single load level of 255 MW (annual peak). A total of 34867 samples are characterized through MCS to create a training dataset with 33744 successes and 1123 failure states. In these samples a total of 2500 states are selected to train the classifier. After the training, the system is tested until COV reaches the limit of ≤ 1%. During this phase, 301583 samples are classified with 291243 success and 10340 failure states. The overall performance of MLKNN classifier is stated in table I as well as simulation times for each model (MLKNN Classifier and OPF). The classification performance is compared to results obtained from DC-OPF analyses in table II. The performance indices to present obtained results are calculated according to metrics described before.

According to the results stated in table 3.1, MLKNN Classifier can successfully identify overall LOLP of IEEE 30 Bus Test system with a small error rate. Table 3.1 also shows that MLKNN classification reduces the computation time for calculating reliability indices dramatically comparing to OPF based MCS methods.

Table 3.2 shows that MLKNN classification method provides reasonably accurate classification of bus states of RTS composite power system (success or failure).

Table 3.1: Comparison of Overall System Performance for MLKNN & CMCS for IEEE 30 Bus Test System

	CMCS	MLKNN
Success States	291792	291502
Failure States	9791	10081
Loss of Load	0.032	0.033
Sensitivity	N/A	0.99812
Specificity	N/A	1
G-Mean	N/A	0.999
Analysis Time (Sec)	28659	1639

2. Performance on IEEE RTS

In case study two, to test performance of the proposed classifier, the system is tested on single area IEEE RTS for single load level described as 2850 MW (annual peak). There are 10 generation buses in RTS which are considered as input vector. To train classifier, 14682 samples are obtained through MCS with 13381 successes and 1301 failure in this process. After obtaining adequate number of samples, the training patterns are recombined to generate a balanced training dataset (some of the success states are discarded to prevent overtraining). A total of 3000 samples are selected with 1301 failure and 2699 success states for this dataset. It should be noted that most of the success states are ignored during this process to emphasize classification of failure states (which is reasonable in order to calculate reliability indices). After MLKNN classifier is successfully trained, the proposed system is tested until COV reaches the limit of $\leq 1\%$. After testing is completed, 109743 samples are classified with 100470 successes and 9273 failure states.

Table 3.2: Comparison in Bus Level for IEEE 30 Bus Test System

	CMCS		MLKNN		Accuracy		
	FN	**LOL (%)**	**FN**	**LOL (%)**	**Sensitivity**	**Specificity**	**G-Mean**
Bus1	0	0.00	0	0.00	N/A	N/A	N/A
Bus2	1712	0.57	1831	0.61	0.998	0.988	0.9929
Bus3	1519	0.50	1511	0.50	0.999	0.984	0.9914
Bus4	3379	1.12	3363	1.12	1	0.993	0.9964
Bus5	3558	1.18	3562	1.18	0.999	0.996	0.9974
Bus6	0	0.00	0	0.00	N/A	N/A	N/A
Bus7	2187	0.73	2194	0.73	0.999	1	0.9995
Bus8	2030	0.67	2022	0.67	0.999	0.991	0.9949
Bus9	0	0.00	0	0.00	N/A	N/A	N/A
Bus10	1049	0.35	1038	0.34	1	0.981	0.9904
Bus11	0	0.00	0	0.00	N/A	N/A	N/A
Bus12	2366	0.78	2348	0.78	0.999	0.986	0.9924
Bus13	0	0.00	0	0.00	N/A	N/A	N/A
Bus14	2747	0.91	2729	0.90	1	0.989	0.9944
Bus15	1686	0.56	1671	0.55	0.999	0.985	0.9919
Bus16	1068	0.35	1045	0.35	0.999	0.974	0.9864
Bus17	784	0.26	771	0.26	1	0.971	0.9853
Bus18	1412	0.47	1394	0.46	0.999	0.982	0.9904
Bus19	1500	0.50	1486	0.49	0.999	0.982	0.9904
Bus20	1436	0.48	1423	0.47	0.999	0.981	0.9899
Bus21	1755	0.58	1868	0.62	0.999	0.985	0.9919
Bus22	0	0.00	0	0.00	N/A	N/A	N/A
Bus23	1313	0.44	1291	0.43	1	0.977	0.9884
Bus24	1370	0.45	1358	0.45	1	0.983	0.9914
Bus25	0	0.00	0	0.00	N/A	N/A	N/A
Bus26	1194	0.40	1182	0.39	1	0.976	0.9879
Bus27	0	0.00	0	0.00	N/A	N/A	N/A
Bus28	0	0.00	0	0.00	N/A	N/A	N/A
Bus29	469	0.16	451	0.15	1	0.958	0.9787
Bus30	1464	0.49	1447	0.48	1	0.981	0.9904

Table 3.3: Comparison of Overall System Performance for IEEE RTS

	CMCS	**MLKNN**
Success States	100470	100402
Failure States	9273	9341
Loss of Load Probability	0.0845	0.085
Sensitivity	N/A	0.997
Specificity	N/A	0.993
G-Mean	N/A	0.994
Analysis Time (Sec)	6271	448

The overall classification performance of MLKNN classifier and time comparison between the proposed method and CMCS is stated in table 3.3. The classification performance at bus level is stated with comparison of results obtained from DC-OPF analyses in table 3.4.

According to the results stated in table III, MLKNN Classifier can successfully identify overall LOLP of RTS system with a small error rate. Similar to Case study one, MLKNN classification reduces computation time of reliability indices significantly comparing to OPF based CMCS.

As shown in Table IV MLKNN classification method provides reasonably accurate classification of bus states of RTS composite power system (success or failure). It should be noted that if failure rate at a bus is very small (like bus-9) classifier may show lower performance than average. The reason is that, there are not enough samples generated to adjudicate properly for those buses neither in training nor testing stages.

Table 3.4: Comparison of MLKNN & CMCS in Bus Level

	MLKNN		CMCS		Accuracy		
	FN	**LOL (%)**	**FN**	**LOL (%)**	**Sensitivity**	**Specificity**	**G-Mean**
Bus1	2973	2.71	2934	2.67	0.998	0.974	0.9859
Bus2	2937	2.68	2953	2.69	0.999	0.981	0.9899
Bus3	0	0.00	0	0.00	N/A	N/A	N/A
Bus4	0	0.00	0	0.00	N/A	N/A	N/A
Bus5	936	0.85	952	0.87	0.999	0.972	0.9854
Bus6	33	0.03	29	0.03	0.999	0.896	0.9460
Bus7	5032	4.59	5063	4.61	0.998	0.981	0.9894
Bus8	648	0.59	642	0.59	0.999	0.986	0.9924
Bus9	3	0.0	1	0.0	0.999	0.33	0.5741
Bus10	20	0.02	21	0.02	0.999	0.952	0.9752
Bus11	0	0.00	0	0.00	N/A	N/A	N/A
Bus12	0	0.00	0	0.00	N/A	N/A	N/A
Bus13	456	0.42	471	0.43	0.999	0.946	0.9721
Bus14	208	0.19	224	0.20	0.999	0.902	0.9492
Bus15	0	0.00	0	0.00	N/A	N/A	N/A
Bus16	56	0.05	51	0.05	1	0.882	0.9391
Bus17	0	0.00	0	0.00	N/A	N/A	N/A
Bus18	1018	0.93	1034	0.94	0.999	0.907	0.9518
Bus19	48	0.04	51	0.05	0.999	0.921	0.9592
Bus20	3802	3.46	3879	3.53	0.997	0.946	0.971
Bus21	0	0.00	0	0.00	N/A	N/A	N/A
Bus22	0	0.00	0	0.00	N/A	N/A	N/A
Bus23	0	0.00	0	0.00	N/A	N/A	N/A
Bus24	0	0.00	0	0.00	N/A	N/A	N/A

3. Performance on Varying Hourly Load

In case study three, performance of the proposed method is tested on varying load levels based on information provided in hourly load chart of RTS. To be able to classify varying load levels, load information should be added to the input of the classifier. For this purpose, the input equation described in (3.1) is modified and new equation is described below in (3.23).

$$I_{input} = \begin{bmatrix} G_{11} - D_{11} & G_{12} - D_{12} & G_{1N} - D_{1N} \\ G_{21} - D_{21} & G_{22} - D_{22} & G_{2N} - D_{2N} \\ G_{M1} - D_{M1} & G_{M2} - D_{M2} & G_{MN} - D_{MN} \end{bmatrix} \tag{3.23}$$

In this equation G represents generation and D represents load at bus N for total for M number of samples.

However, the size of classifier can become too large while classifying system states in varying load levels which reduces computational efficiency of classifier. In this study, instead of using one classifier, multiple classifiers are trained for different load levels to handle this problem efficiently. Total state space sampled is divided to five main level based on available generation data. For each of those levels a unique classifier is trained. In the testing stage, a decision tree is used to determine which classifier to be used for classification for every random sample. The overall diagram of the algorithm used in this case study is given in figure 3.3.

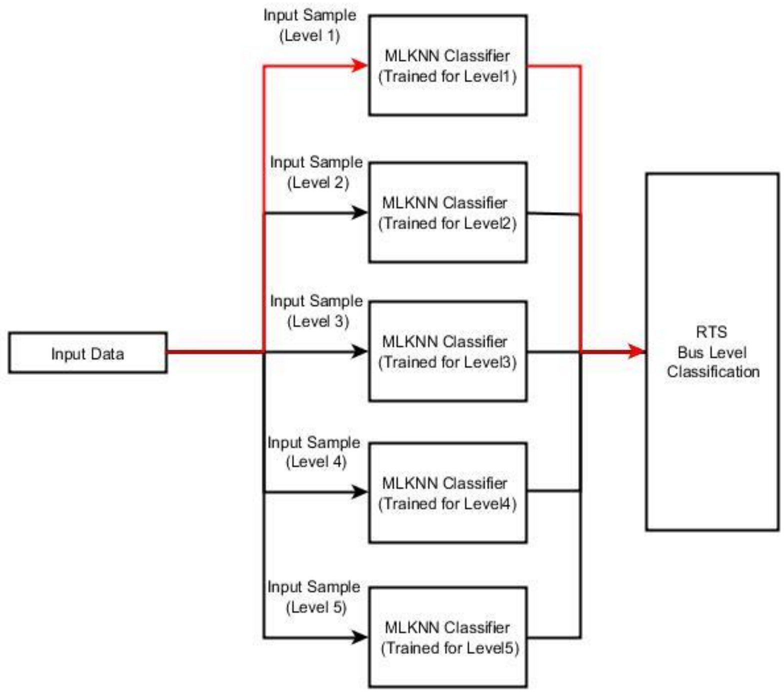

Figure 3.3: Overall Diagram of Proposed Classifier for Variable Load Levels

In the training phase of classifier, MCS is run for each load level until a total of 5000 samples are obtained with 3000 success and 2000 failure states. Most of the success states are discarded to prevent overtraining as in the previous case studies. After MLKNN classifier is successfully trained, the proposed system is tested until COV reaches the limit of $\leq$ 1%. After testing is completed, 7540967 samples are classified with 7540967 successes and 9052 failure states.

The overall classification performance of proposed method for characterizing random samples is given in table 3.5. The classification performance at bus level is presented with comparison of results obtained from DC-OPF analyses in table 3.6.

Results obtained in table 3.5 show that the proposed classifier can successfully identify overall LOLP of RTS system with an acceptable error rate. The results also show that computation time of computing reliability indices are reduced significantly compared to CMCS. As presented in Table 3.6 MLKNN classifier can characterize system buses very accurately in varying load levels.

Table 3.5: Comparison of System Performance for Varying Hourly Load

	CMCS	**MLKNN**
Success States	7540967	7540881
Failure States	9052	9138
Loss of Load	0.0012	0.0012
Sensitivity	N/A	0.99
Specificity	N/A	0.96
G-Mean	N/A	0.97
Analysis Time (Sec)	375249	33492

Table 3.6: Comparison of System Performance for Varying Hourly Load in Bus Level

Location	CMCS (Failures)	MLKNN (Failures)	Sensitivity	Specificity	G-Mean
Bus1	2115	2178	0.99	0.97	0.98
Bus2	1157	1204	0.99	0.98	0.98
Bus3	N/A	N/A	N/A	N/A	N/A
Bus4	N/A	N/A	N/A	N/A	N/A
Bus5	606	612	0.99	0.97	0.98
Bus6	36	51	0.99	0.92	0.95
Bus7	4143	4051	0.99	0.91	0.95
Bus8	325	371	0.99	0.99	0.99
Bus9	N/A	N/A	N/A	N/A	N/A
Bus10	N/A	N/A	N/A	N/A	N/A
Bus11	N/A	N/A	N/A	N/A	N/A
Bus12	N/A	N/A	N/A	N/A	N/A
Bus13	269	291	0.99	0.93	0.96
Bus14	247	278	0.99	0.91	0.95
Bus15	0	0	N/A	N/A	N/A
Bus16	43	49	1	0.91	0.95
Bus17	0	0	N/A	N/A	N/A
Bus18	1797	1658	1	0.86	0.93
Bus19	239	291	1	0.94	0.97
Bus20	4484	4303	1	0.93	0.96
Bus21	N/A	N/A	N/A	N/A	N/A
Bus22	N/A	N/A	N/A	N/A	N/A
Bus23	N/A	N/A	N/A	N/A	N/A
Bus24	N/A	N/A	N/A	N/A	N/A

4. Performance Including Transmission Line Failures

In case study four, the proposed method is applied to classify transmission line failures. To create proper patterns to classify transmission line failures, states of transmission lines should be combined with the generation data as an input of classifier. Many of the previous studies that have a focus on machine learning techniques for state classification of composite power systems ignores system failures sourced from transmission lines by using some assumptions. The main difficulty lies here is combining information of available generation and transmission line capacity at the input of the classifier to create proper patterns. In this study, a new approach is proposed to achieve classification of failures reasoned from transmission line failures. In this approach, equation (3.1) is modified by applying discrete time convolution to the information obtained from generation and transmission line states. New input equation is described below in (3.24) below.

$$I_{input} = \begin{bmatrix} G_{11} & G_{12} & G_{1N} \\ G_{21} & G_{22} & G_{2N} \\ G_{M1} & G_{M2} & G_{MN} \end{bmatrix} * \begin{bmatrix} Tr_{11} & Tr_{12} & Tr_{1L} \\ Tr_{21} & Tr_{22} & Tr_{2L} \\ Tr_{M1} & Tr_{M2} & Tr_{ML} \end{bmatrix} \qquad (3.24)$$

where G represents generation at bus N and Tr represents available transmission line capacity for transmission line L at total M number of samples. The description of discrete time convolution which is symbolized as ‘*’ is presented in equation (3.25).

$$(G * Tr) = \sum_m G(i-m) x Tr(m) \qquad (3.25)$$

In the training phase, MCS is run for each load level until a total of 10000 samples are obtained with 5000 failure and 5000 success states. As in the previous case studies, the remaining success states are discarded to prevent overtraining. After MLKNN classifier is successfully trained, the proposed system is tested until COV reaches the limit of $\leq 1\%$. After testing is completed, 200387 samples are classified with 189659 successes and 10728 failure states. It should be noted that load level is considered as constant at its annual peak level.

The overall classification performance of the proposed method for characterizing random samples is given in table 3.7. The classification performance at bus level is presented with comparison of results obtained from DC-OPF analyses in table 3.8. Finally, table IX demonstrates the classification performance of the proposed approach only for system failures sourced from transmission line failures.

Results obtained in table 3.7 show that the proposed classifier can successfully identify overall LOLP of RTS system with an acceptable error rate. The results also show that computation time of computing reliability indices are reduced significantly comparing to CMCS. As presented in Table 3.8 MLKNN classifier can characterize system buses very accurately when transmission line failures considered.

Table 3.7: Performance Comparison when Transmission Line Failures Considered

	CMCS	MLKNN
Success States	190659	190583
Failure States	9728	9804
Loss of Load	0.0485	0.0489
Sensitivity	N/A	0.99
Specificity	N/A	0.98
G-Mean	N/A	0.99
Analysis Time (Sec)	12390	637

It is shown at table 3.9 that overall system failures sourced from transmission line failures are characterized with a high accuracy. The same table also shows that proposed method has a good classification accuracy for bus level characterization.

Table 3.8: Comparison of Bus Level Classification Performance for Case 4

	CMCS	MLKNN	Sensitivity	Specificity	G-Mean
Bus1	961	986	0.98	0.97	0.97
Bus2	304	321	0.99	0.98	0.98
Bus3	42	51	N/A	N/A	N/A
Bus4	194	206	N/A	N/A	N/A
Bus5	360	372	0.99	0.97	0.98
Bus6	2003	2028	0.99	0.92	0.95
Bus7	1282	1307	0.99	0.97	0.98
Bus8	440	452	0.99	0.99	0.99
Bus9	38	47	0.99	0.98	0.98
Bus10	258	271	0.99	0.97	0.98
Bus11	N/A	N/A	N/A	N/A	N/A
Bus12	N/A	N/A	N/A	N/A	N/A
Bus13	1623	1702	0.99	0.93	0.96
Bus14	4929	5013	0.99	0.91	0.95
Bus15	1	3	N/A	N/A	N/A
Bus16	821	833	1	0.91	0.95
Bus17	N/A	N/A	N/A	N/A	N/A
Bus18	721	739	0.99	0.89	0.94
Bus19	320	332	0.99	0.93	0.96
Bus20	400	421	0.99	0.97	0.98
Bus21	N/A	N/A	N/A	N/A	N/A
Bus22	N/A	N/A	N/A	N/A	N/A
Bus23	N/A	N/A	N/A	N/A	N/A
Bus24	N/A	N/A	N/A	N/A	N/A

Table 3.9: Classification Performance for System Failures Sourced from Transmission Line Failures

	CMCS	MLKNN	Sensitivity	Specificity	G-Mean
Bus1	10	13	0.98	1	0.99
Bus2	N/A	N/A	N/A	N/A	N/A
Bus3	22	22	1	1	1
Bus4	34	36	0.99	0.97	0.98
Bus5	26	23	0.97	0.85	0.91
Bus6	127	123	0.98	0.95	0.96
Bus7	7	8	0.99	0.88	0.93
Bus8	44	46	0.98	0.96	0.97
Bus9	17	15	1	0.88	0.94
Bus10	58	57	1	0.98	0.99
Bus11	N/A	N/A	N/A	N/A	N/A
Bus12	N/A	N/A	N/A	N/A	N/A
Bus13	3	3	1	1	1
Bus14	42	45	0.99	0.98	0.99
Bus15	24	24	1	1	1
Bus16	14	15	1	0.91	0.9391
Bus17	N/A	N/A	N/A	N/A	N/A
Bus18	N/A	N/A	N/A	N/A	N/A
Bus19	18	18	1	1	1
Bus20	16	16	1	1	1
Bus21	N/A	N/A	N/A	N/A	N/A
Bus22	N/A	N/A	N/A	N/A	N/A
Bus23	N/A	N/A	N/A	N/A	N/A
Bus24	N/A	N/A	N/A	N/A	N/A
Overall System	310	307	1	0.99	0.99

F. Conclusion

This chapter has presented a method to evaluate reliability indices for composite power systems. The proposed method uses a MLKNN classifier to identify status of buses that does significantly reduce the computational burden of OPF analysis. The importance of reducing the computational time can be understood by two examples. In Monte Carlo Simulation, the accuracy of convergence is very important. Convergence is measured by the COV, smaller COV means better convergence.

Now the sample size (or computational time) is inversely proportional to COV or directly proportional to accuracy of convergence. The proposed method reduces the required time for reliability analysis considerably for the same level of accuracy defined by the coefficient of variation. Alternatively, for the same time this allows convergence to a higher level of accuracy. Another example is that for optimal planning of resources, reliability studies may need to be done many times. So, the reduction of computational time helps in optimal planning.

The effectiveness of proposed method is demonstrated on the IEEE 30 Bus Test System and IEEE RTS respectively in four different case studies. In first two case studies, the load level of system is considered constant at its peak value, in the third case study performance of proposed method is tested on varying hourly load data of RTS and finally, the proposed system is applied on MRTS with considering transmission line failures. The accuracy of classification is evaluated by considering the parameter sensitivity, specificity and g-mean. The training samples are chosen with CMCS. After classifier is trained, testing is completed with CMCS until COV reaches 1% limit. All classifications in this step are made by MLKNN instead of DC-OPF analysis.

The results show that the proposed method shows good performance for classifying success and failure states at constant load level for both IEEE 30 bus system and RTS. In some buses with lower failure rate, however, classification accuracy performs slightly below of average in some cases. This could be amended by adding more samples to the training dataset. It should also be mentioned that, classification of success states showed better performance than classification of failure states.

In the third case study, demand information of sampled state is included to the input of the classifier for characterizing system state in varying load levels. In this stage, a decision tree is applied to choose one of the five different classifiers which are trained with a focus of load level of sampled states. The simulation results presented demonstrate that the proposed method can execute the classification with a good accuracy.

In the last case study, a new approach is introduced to handle classification of failures sourced from inadequate transmission line capacities. Results presented for this case study show that the proposed approach can characterize these states with a high accuracy.

It is also shown in all case studies that, the time required for calculating composite system reliability indices with MLKNN classifier is much less than OPF based reliability evaluation methods. These case studies demonstrate that the application of the proposed method on composite power system reliability evaluation accurately determines the states status with a huge speed up compared with OPF based Monte Carlo Simulation methods. This method advances the state of the art of using machine learning in power system reliability evaluation from the previous methods by including computation of bus indices and the transmission line failures.

CHAPTER IV

MULTI-LABEL CLASSIFICATION AND IMPORTANCE SAMPLING COMBINATION FOR COMPOSITE SYSTEM RELIABILITY EVALUATION

In this chapter a new approach for evaluation of power systems reliability indices with Monte Carlo Simulation is presented with a combination of Multi-label Radial Basis Function (MLRBF) classifier and Importance sampling (IS). Multi-label classification algorithms is different from single label approaches, in which each instance can be assigned into multiple classes. This characteristic gives MLRBF a capability to be used to classify composite power system states (success or failure) without requiring optimal power flow (OPF) analysis, with the exception of training and cross-entropy optimization phases. The proposed method is applied to the IEEE RTS for different load level scenarios. The outcomes of case studies show that MLRBF algorithm together with importance sampling provides good classification accuracy in reliability evaluation while reducing computation time substantially. The details of proposed method are explained followingly.

A. MLRBF Classification for Power System Reliability Evaluation

RBF is one of the most popular techniques among neural network classification methods. RBF Neural Networks are generally comprised of two layers of neurons. In RBF, each hidden neuron (basis function) in the first layer is associated with a prototype vector while each output neuron corresponds to a possible class. Usually training an RBF neural network is handled in a two-stage procedure. In the first layer, the basis functions are learned by performing clustering analysis on training instances while weights are optimized by solving a linear problem in the second layer. Comprehensive descriptions of RBF neural networks are available in [32]. In this section, first, a general formulation of composite system reliability evaluation parameters for MLRBF classification is explained, later, application of the proposed method is described in steps. Finally, a flow chart of MLRBF is provided for clearer understanding in Figure 4.1.

1. General Definitions for MLRBF in Power System Reliability Evaluation

In this study, total generation capacities and total demand for each bus of composite system are taken as input parameters for MLRBF classifier. So, generation and demand information for each bus in the system is considered as an element of input matrix I for every sample (instance) m as described in (4.1).

$$I_{input} = \begin{bmatrix} G_{11} - D_{11} & G_{12} - D_{12} & G_{1N} - D_{1N} \\ G_{21} - D_{21} & G_{22} - D_{22} & G_{2N} - D_{2N} \\ G_{M1} - D_{M1} & G_{M2} - D_{M2} & G_{MN} - D_{MN} \end{bmatrix} \tag{4.1}$$

where N is the number of the buses and M is the total number of

samples in the input matrix. Status of state information for each bus of the system for M different samples is stored in a target matrix T for the purpose of training the MLRBF classifier which described in (4.2).

$$T = \begin{bmatrix} S_{11} & S_{12} & S_{1Q} \\ S_{21} & S_{22} & S_{2Q} \\ S_{M1} & S_{M2} & S_{MQ} \end{bmatrix} \tag{4.2}$$

where Q is the number of the load buses in the system and S is the status information of bus q. While defining status of buses '-1' is used for 'success states' and '1' for 'failure states' for the corresponding bus.

P_{out}, contains failure probability for each bus of composite system for each sample M as the output for this classifier which described in (4.3).

$$P_{out} = \begin{bmatrix} P_{11} & P_{12} & P_{1Q} \\ P_{21} & P_{22} & P_{2Q} \\ P_{M1} & P_{M2} & P_{MQ} \end{bmatrix} \tag{4.3}$$

Now, training and testing procedure is explained in steps in following subsection.

2. *Explanation of MLRBF Classification Procedure*

It is necessary to describe some related parameters before starting explanation;

m: defines index of current sample of total M samples.

i_m*:* defines the input vector for sample m.

q: defines the bus index of total Q buses of system.

Y_m defines the state of bus *q* in sample *m* so;

$$Y_m(q) = \begin{cases} 1\ (failure)\ where\ \ T_{iq} = 1 \\ 0\ (success)\ where\ T_{iq} = -1 \end{cases}.$$

Let $I=R^d$ be the input space and Q= {1, 2…, Q} be the finite set of Q possible classes. Given a multi-label training dataset DSet= $\{(i_m,Y_m)| \leq m \geq M\}$, where $i_m \in I$ is a single instance and $Y_m \subseteq Q$ is label set associated with i_m.

In this study, K-Means Clustering is applied for each class $q \in Q$ on the set of instances U_q with label q which described in (4.4).

$$U_q= \{i_m \,|(i_m Y_m) \in DSet,\, q \in Y_m\} \tag{4.4}$$

In the next step, k_q number of clustered groups are formed for class q and the jth centroid ($1\leq j\leq k_q$) is regarded as a prototype vector $c^q{}_j$ of basis function $\alpha^l{}_j(.)$. It should be noted that, k_q is taken as a fraction of the total number of instances in U_q which is described as α.

As each output neuron of the MLRBF neural network is related to a possible class, weights between hidden and output layer can be shown as (4.5).

$$W=[w_{jq}]_{(K+1)\, XQ} \tag{4.5}$$

Here, $K = \sum_{q=1}^{Q} k_q$ shows the total number of prototype vectors retained in the hidden layer. The weight matrix W can be learned by minimizing the following sum-of-squares error function as described below (4.6).

$$E = \frac{1}{2}\sum_{m=1}^{M}\sum_{q=1}^{Q}\left(Y_q(i_m) - T_q^m\right)^2 \tag{4.6}$$

Where T_q^m represents the output of i_m on the q-th class, which takes the values of +1 if q $\in Y_i$ and -1 otherwise. So, the output of i_m for the *q*-th class can be calculated as presented below (4.7).

$$y_q(i_m) = \sum_{j=0}^{Q} w_{jq}\phi_j(i_m) \tag{4.7}$$

In this study, the basis function a_j is represented with the following widely-used Gaussian style activation (4.8).

$$\phi_j(i_m) = \exp(-\frac{dist(i_m,c_j)^2}{2\sigma_j^2}) \tag{4.8}$$

Here dist (i_m,c_j) calculates the distance between i_m and the *j*-th prototype vector c_j with the usual Euclidean distance algorithm. The smoothing parameter σ is shown with the equation below (4.9).

$$\sigma = \left(\frac{\sum_{p=1}^{K-1}\sum_{r=p+1}^{K} dist(c_p,c_r)}{\frac{K(K-1)}{2}}\right) \tag{4.9}$$

Differentiating the error function (4.6) with respect to w_{jq} and setting the derivative to zero be results with the equation given below (4.10).

$$(\phi^T \phi)W = \phi^T T \tag{4.10}$$

In equation 4.10, $\phi = [\phi_{mj}]_{m \times [K+1]}$ with elements, $\phi_{mj} = \phi_j(i_m)$, W= $[w_{jq}]_{Q \times [K+1]}$ and T=$[t_{mq}]_{m \times Q}$ with elements $t_{mq} = t_q^m$.

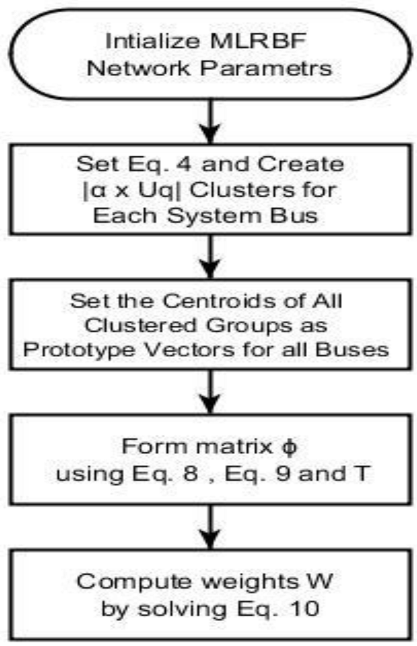

Figure 4.1: Flowchart Describing Training Phase of MLRBF

B. Importance Sampling

Importance sampling is one of the most successful variance reduction techniques used in reliability evaluation of composite power systems [34]. IS changes the probability density function of occurrences by emphasizing certain values of a random variable which have greater impact, when compared with others, on the estimation process of a target quantity. Consequently, values which have more importance are sampled more often and the variance of the estimator is reduced faster. IS aims to select a probability density function different from the original to minimize variance of samples [35]. To be able to obtain the maximum performance from importance sampling, selected probability density function should be equal or close to optimum $f_{optimum}(.)$ which initially unknown.

At this stage the CE method can be utilized for estimating the optimal, or at least close to optimal, reference parameters by minimizing the distance between the original sampling density and the optimal sampling density $f_{optimum}(.)$ iteratively.

Detailed technical information can be obtained from [36]. In the following subsections general definitions regarding IS are presented and later implementation of CE method for power system reliability evaluation is described.

1. General Definition of Importance Sampling

Consider a power system with G_N generating stations. Also, assume that the system has J identical and independent units, each one with a capacity G_{nj} for each of N stations. Let u_{nj} be a vector which contains original probability of unavailability of all generation units in the system. Under this assumption, the analytical problem of evaluating the LOLP index can be described by the following equation:

$$LOLP = \frac{1}{M}\sum_{i=1}^{M} H(X_i) \tag{4.11}$$

Where X_i represents ith sample of M total samples and H represents the test function which takes value of 1 if sample X_i has loss of load and 0 otherwise. Under these assumptions IS can be applied to the system by using the new unviability vector v_{nj} to calculate H_{IS}.

$$LOLP_{IS} = \frac{1}{M_{Is}} \sum_{i=1}^{M_{Is}} H_{IS}(X_i) W(X_i; m_{IS}; u; v) \tag{4.12}$$

The expression $W(X_i; m_{IS}; u; v)$ is called likelihood ratio. This value represents a necessary correction in the sampling process because of the changed unavailability vector v. In this study, $W(X_i; m_{IS}; u; v)$ is calculated by (4.13).

$$W(X_i; m_{IS}; u; v) = \frac{\prod_{j=1}^{N_G} (1-u_j)^{x_j^G} (u_j)^{1-x_j^G}}{\prod_{j=1}^{N_G} (1-v_j)^{x_j^G} (v_j)^{1-x_j^G}} \tag{4.13}$$

Where x_j^G represents the availability of generation unit j. The main problem in this process is defining optimal v values to minimize computation time. In this study CE algorithm is utilized for this purpose which is explained in following subsection.

2. CE Algorithm

In this subsection, CE algorithm to determine optimal unavailability (v) values for each generation unit is described. Detailed information about CE can be found in [37].

The algorithm used in this study converges to optimal v parameters using an iterative procedure. During each iteration, v parameters are updated by using predefined number of system state samples. CE algorithm contains 6 main steps. While optimal v parameters are estimated in steps (1-4), loss of load indices are calculated with IS-MCS in steps (5-6).

1- Define the initialization parameters as sample size used for each iteration N (e.g. 50,000) and multilevel parameter p (e.g. between 0.01 and 0.1). Define v_0=u, t=1 and $\phi = L_{LMAX}$ where v represents the updated unavailability vector, t iteration number, ϕ stopping criteria for performance function and L_{LMAX} maximum peak load of the system.

2- Generate a set of random samples of states $X_1, X_2 \ldots X_N$ from the densities f (., v_{t-1}). Evaluate the performance of selected states S= [S_1, S_2 …, $S(X_N)$] according to the selected performance function. Sort the performances of the states in an increasing order so that $S_{[1]} \leq S_{[2]} \leq \ldots \leq S_{[XN]}$ then compute the performance of state (p) quantile of the performances, $S_{[(p)XN]}$.

3- Set the $\phi_t = S_{[(1-p)\,XN]}$ provided that ϕ_t is less than ϕ otherwise set $\phi_t = \phi$. Evaluate the indicator function $H(X_i)$ such that $H(X_i)=1$ if $S(X_i) > \phi_t$ otherwise $H(X_i)=0$.

4- Use the sample from step 2 to update the new unviability vector

$$v_{tj} = 1 - \frac{\sum_{i=1}^{X_N} H_t(X_i) W_{i,t-1} X_{ij}}{\sum_{i=1}^{X_N} H_t(X_i) W_{i,t-1}} \quad (4.14)$$

Where

$$W_{i,t-1} = W(X_i; u; v_{t-1}) \quad (4.15)$$

5- If $\phi_t = \phi$ criteria has been satisfied optimal parameters has been found otherwise increase iteration number as t=t+1 and go back to the step 2.

6- Calculate loss of load indices with the equation (12) by using the optimal parameters defined in step 5.

A flowchart is provided for a clear understating in explaining cross-entropy method at figure 4.2.

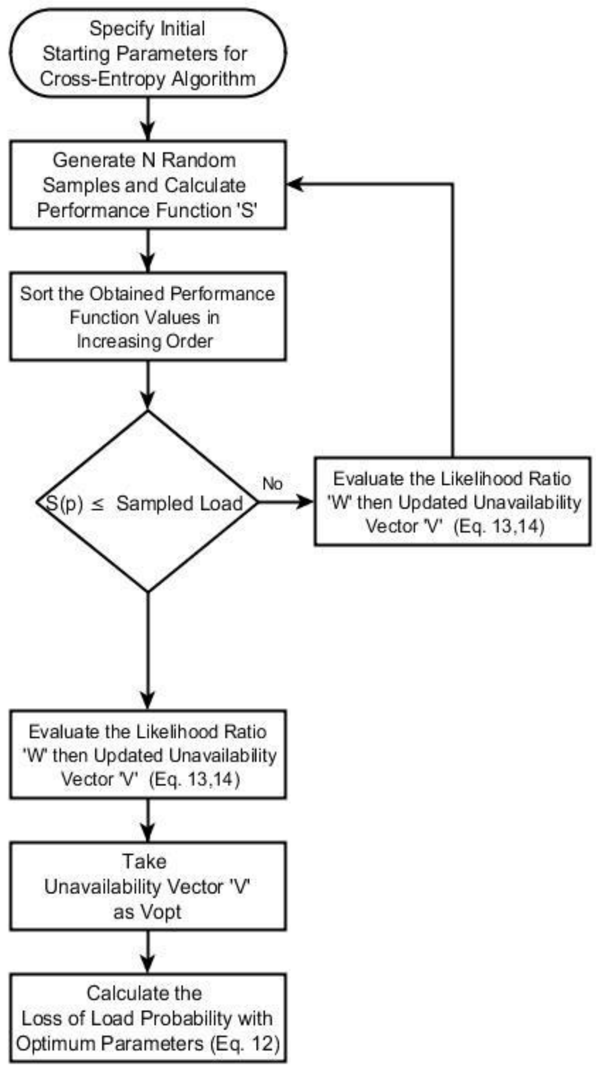

Figure 4.2: Flow Chart of Importance Sampling

C. Application Procedure of Proposed Method

In this section application procedure of the proposed method is explained. This novel approach calculates reliability indices of composite power systems by combining multi-label classifier and importance sampling technique within the framework of Monte Carlo Simulation (MCS). Generally, the most time-consuming part of composite system reliability evaluation is the optimal power flow analysis (OPF). This approach proposes use of a faster Multi-label classifier instead of OPF analysis after proper training. The proposed method can be applied while using either nonsequential or sequential MCS however nonsequential approach is chosen to illustrate performance in this study because of simpler architecture. A benchmark created with Crude Monte Carlo Simulation (CMCS) analysis is also provided for comparison purpose.

The first step of applying combination of IS and ML classification is determining optimum unavailability vector via CE method as described in section B. After optimum unavailability vector is created then multi-label classifier is trained and tested to use state space classification for evaluating the reliability indices of composite power system. Detailed implementation of multi-label classifier is described below in two subsections defined as training and testing process, below.

1. Training Process

Training data sampled for this study are created through MCS. The generation and demand information for each bus of the selected sample states is used to create input vector while status of each bus for those states are recorded as target vector as shown in (1-2). To increase training performance input vector variables are normalized between -1 and 1. In this step, failure and non-failure status observed for each bus are labeled as 1 and -1 respectively. It should also be noted that an equal amount of success and failure states is chosen to create training dataset to prevent overtraining.

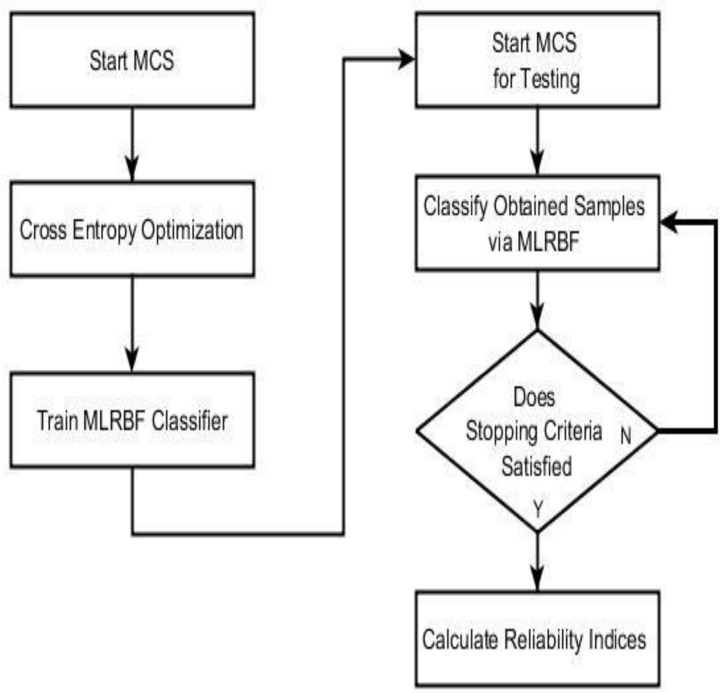

Figure 4.3: Overall Diagram of Proposed Method

2. Testing Process

In the testing process, MCS with importance sampling is used for generating random samples and those states are classified by multi-label classifier until simulation reaches a previously determined stopping criterion. Stopping criteria for this study is defined as the coefficient of variation (COV) to represent the estimated uncertainty.

Reliability indices are calculated and a comparison is made with the results obtained from Crude MCS and IS benchmarks. Reliability indices evaluated are based on Loss of Load Probability (LOLP). Complete flowchart of the proposed methodology is presented in figure 4.3.

D. Case Studies and Results

The IEEE Reliability Test System (RTS) is chosen for demonstration of the proposed method. Two case studies are implemented to demonstrate the performance of the proposed method. In the first case study, load level of RTS is considered constant at its annual peak. In the second case study hourly load data of RTS is divided into 5 different load levels by considering their occurrence probabilities similar to [38]. After the application procedure described in previous section is completed, performance comparison of the proposed multi-label classifier is made with results obtained via CMCS benchmark and standard importance sampling process after calculation of reliability indices for all system buses. Since system losses caused from transmission line failures are much lower than the ones occurring from generation unit failures, states of transmission lines are considered as available at all the time. The capacities of transmission lines are, however, considered. Initial parameters of cross entropy optimization are determined as sample size N=50000, multilevel parameter p=0.05.

All simulations of this study are performed using MATLAB (2012) planform on a PC with Intel Core i7-4510 CPU (~2.6GHz), 16 GB Memory. It should be noted that the results presented below are the average of the 10 simulations.

1. Constant Load Level

In this case study, load level of RTS is considered to be constant and at its peak value of 2,850W. To train MLRBF classifier 6000 samples are selected in which 3000 of them are success states and the rest are failure states. Clustering rate α is chosen as 0.25 for this case study which means number of clusters created is equal to one fourth of times of total failures for each bus as it described in (6). After training of MLRBF classifier is completed, MCS is simulated until COV reaches 1% for all simulation types as it is specified for stopping criteria of testing phase. In this process a total of 109,743 samples were obtained with 100459 successes and 9284 failures characterized.

The comparison of obtained results is made in Tables 1 and 2. Simulation results for overall system classification performance and time comparison of MLRBF Classifier on RTS are presented in Table 4.1. The simulation results for bus level classification performance are stated in Table 4.2. In this study, performance comparison in this study is made based on Loss of Load Probability (LOLP).

Table 4.1 shows that MLRBF classification method can compute overall LOLP of RTS with a small fraction of error and computational time required to evaluate the reliability indices can be significantly reduced by the proposed MLRBF – IS combination when compared to standard MCS methods.

Table 4.1: Comparison on Overall Performance Analysis and CPU Time Spend

Algorithm	**Success States**	**Failure States**	**LOLP $\times 10^{-2}$**	**CPU Time (Sec $\times 10^{3}$)**
CMCS	100459	9284	8.46	5.49
CEIS	18786	1748	8.51	1.03
ML-CEIS	21305	1988	8.53	0.069

Table 4.2: Comparation of Classification Performance at Bus Level Based on LOLP

Location	**CMCS (LOLP) $\times 10^{-2}$**	**CEIS (LOLP) $\times 10^{-2}$**	**ML-CEIS (LOLP) $\times 10^{-2}$**
Bus 1	2.67	2.71	2.82
Bus 2	2.69	2.74	2.84
Bus 3	0.00	0.00	0.00
Bus 4	0.00	0.00	0.00
Bus 5	0.87	0.84	0.89
Bus 6	0.03	0.05	0.06
Bus 7	4.61	4.72	4.75
Bus 8	0.59	0.6	0.68
Bus 9	0.00	0.00	0.00
Bus 10	0.02	0.01	0.04
Bus 11	N/A	N/A	N/A
Bus 12	N/A	N/A	N/A
Bus 13	0.43	0.47	0.52
Bus 14	0.20	0. 24	0.29
Bus 15	0.00	0.00	0.00
Bus 16	0.05	0.1	0.15
Bus 17	N/A	N/A	N/A
Bus 18	0.94	1.1	0.81
Bus 19	0.05	0.12	0.02
Bus 20	3.53	3,59	3.76
Bus 21	N/A	N/A	N/A
Bus 22	N/A	N/A	N/A
Bus 23	N/A	N/A	N/A
Bus 24	N/A	N/A	N/A

Table 4.2 also shows that proposed method shows reasonably accurate classification on characterizing the failed bus states of RTS.

2. Hourly Varying Load Levels

In this case study load level of the system is chosen randomly from the original load data of RTS. There are 8736 different load levels specified in annual hourly load values of RTS. As in the first case study, samples used for training MLRBF classifier are obtained through MCS. Since size of classifier is one of the most determining factors in classification time, multiple classifiers are trained for different load levels to handle this problem efficiently instead of training one large network. For this purpose, five different thresholds are defined based on available power. For each level a unique classifier is trained. In testing stage, a decision three is used to determine which classifier to be used for classification for every random sample. The overall diagram of the algorithm used in this case study is given in figure 4.4.

For this case study each level is trained by 10000 samples which are obtained through MCS sampled with optimal unavailability vector obtained through the CE process. Similar to the first case study, clustering rate α is selected as 0.25 in this process.

After training is completed, MCS is run until COV reaches 1% as in first case study. In this process a total of 7,442,879 samples were obtained by CMCS with 7,433,754 successes and 9,125 failures characterized.

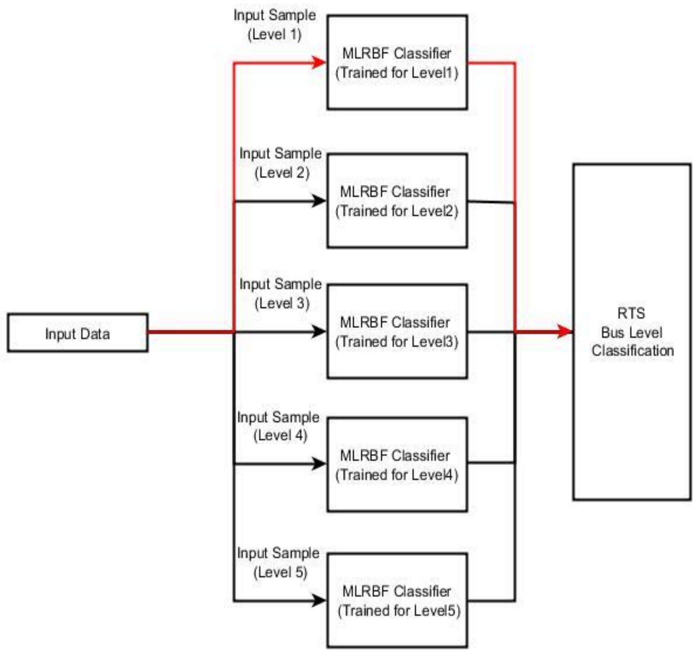

Figure 4.4: Overall Diagram of Proposed Classifier for Variable Load Level

The comparison on classification performance and simulation time for obtained results is presented in Table 4.3 and 4.4. Comparison on overall classification performance and simulation times for proposed method is given in Table 4.3.

Table 4.3: Comparison on Overall Performance Analysis for Varying Load Levels

Algorithm	**Success States**	**Failure States**	**LOLP x 10^{-3}**	**CPU Time (Sec x 10^3)**
CMCS	7433754	9125	1.20	376.690
CEIS	473232	1740	1.23	23.710
ML-CEIS	496277	1817	1.23	1.110

It is clear from Table 4.3 that the proposed method can classify failure states of RTS in multi load level with a close performance. It is also observed in Table 4.3 that the proposed method providing a huge boost in terms of calculation time.

Table 4.4: Comparison of Classification Performance at Bus Level Based on LOLP for Varying Load Levels

Location	**CMCS (LOLP) x 10^{-5}**	**CEIS (LOLP) x 10^{-5}**	**ML-CEIS (LOLP) x 10^{-5}**
Bus 1	28.06	28.7	33.4
Bus 2	15.35	16.43	17.51
Bus 3	0	0.0	0
Bus 4	0	0	0
Bus 5	8.04	8.42	9.14
Bus 6	0.49	0.43	0.21
Bus 7	54.95	54.02	57.42
Bus 8	2.31	1.90	3.28
Bus 9	0	0	0
Bus 10	0	0	0
Bus 11	N/A	N/A	N/A
Bus 12	N/A	N/A	N/A
Bus 13	3.58	3.72	5.97
Bus 14	3.28	3.10	4.51
Bus 15	0	0	0
Bus 16	0.58	0.99	1.2
Bus 17	N/A	N/A	N/A
Bus 18	10.81	12.26	19.58
Bus 19	2.18	2.21	3.84
Bus 20	60.47	61.28	64.91
Bus 21	N/A	N/A	N/A
Bus 22	N/A	N/A	N/A
Bus 23	N/A	N/A	N/A
Bus 24	N/A	N/A	N/A

Table 4.4 shows performance of proposed method for bus level classification. It can be observed from the results that the proposed method can compute the LOLP in close range of CMCS.

E. Conclusion

In this study, a new method is presented to evaluate reliability indices for composite power systems. The proposed method uses a MLRBF classifier to identify status of buses in a way that does not require OPF analysis during Monte Carlo Simulation. The effectiveness of the proposed method is demonstrated on the IEEE RTS.

As can be observed from the results, MLRBF classifier can classify loss of load states with good accuracy most of the times. It should also be noted that rate of classification errors increases in states with low frequency of failures. The main reason of this performance loss for those buses is lack of adequate samples in the training dataset. Performance of the proposed method can be increased by adding more samples to the training dataset as a natural outcome. The main advantage of the proposed method is its ability of reducing the time for reliability analysis considerably which is shown in two different case studies.

CHAPTER V

DEEP LEARNING FOR COMPOSITE POWER SYSTEM RELIABILITY EVALUATION

In many cases, researchers on reliability analysis are interested in finding methods for efficient searching of the state space of the system for states that represent events of interest, like failure of the system to meet the required demand for a set of specific nodes. This indicates the importance for methods that efficiently determine states to be examined and then evaluated. Artificial Intelligence (AI) based methods have been studied either in themselves or in conjunction with widely used methods like Monte Carlo Simulation (MCS) for this objective. In recent years, deep learning techniques have received considerable attention and showed significant promise in many fields when compared to other AI techniques. In this chapter a novel methodology based on combination of deep Convolutional Neural Networks (CNN) and MCS is presented for evaluation of composite power system reliability. This approach is applied to the IEEE Reliability Test System (RTS) by using both AC and DC power flow models for different load levels. The case studies show that the proposed method has a superior performance in both classification accuracy and reducing computational burden of reliability evaluation compared to previous AI based studies.

A. Convolutional Neural Networks

Recently, deep learning algorithms have drawn significant attention in the area of artificial intelligence. This terminology is basically an extension of traditional artificial neural networks (ANN). These algorithms have dramatically improved the state of the art in areas like speech recognition, visual object recognition, object detection and many other domains such as drug discovery and genomics. Deep learning allows computational models that are composed of multiple processing layers to learn representations of data with multiple levels of abstraction. Their immense capability of learning optimal features from raw input data allows avoiding feature engineering. Through these algorithms, pattern classification performance of machines has increased even more than humans in some applications [39, 40]. This chapter explores a new method for composite system reliability evaluation with combination of CNNs and MCS by considering both DC and AC approaches.

CNN is a deep feed forward artificial neural network algorithm which is one of the most used architectures among deep learning methods. It can simply be described as neural networks that use convolution in place of general matrix multiplication in at least one of their layers. CNNs are inspired by research done on the visual cortex of mammals and how they perceive the world using a layered architecture of neurons in the brain, and the overall architecture is reminiscent of the LGN–V1–V2–V4–IT hierarchy in the visual cortex ventral pathway [41,42]. The CNNs can encode certain properties into the architecture which results in less feature engineering requirements compared to other algorithms. Also, CNNs are easier to train and have much fewer parameters than fully connected networks with the same number of hidden units for this reason. In CNNs multi-level neural networks are trained with less neuron requirements. The ability to characterize system features in its own system makes CNNs more suitable for many pattern recognition problems [43].

A typical CNN architecture consists of three stages including convolutional layers, pooling layers and fully connected layers. Input data is sampled into smaller sized feature maps by filters in convolution layer. This process is done by computing the dot product between the entries of the filter and the input. Then pooling layers are applied to reduce the size of the data obtained in convolutional layer. This is followed by connected layer. In this layer activation function is applied to the features gathered in the previous layers, as seen in regular neural networks. At the end, predictions for trained classes can be obtained by applying a softmax function. Remaining of this section describes basic concepts of CNNs in subsections. Rigorous theoretical explanation on CNNs can be found in [44,45].

1. Convolutional Layer

Typically, in convolutional layers input data is applied to a convolutional operation to transfer the results to the next layer. In convolutional network terminology, the first argument to the convolution is often referred to as the input, and the second argument as the kernel or feature map. In usual convolution process, the kernels have flipped relative to the input. This process is not necessary in neural network implementations. Instead, many neural network algorithms implement a related function called as cross-correlation, which has a similar process with convolution but without flipping the kernels. Cross-Correlation operation is described in eq (5.1) for a 1-dimensional input.

$$(Input * Kernel)(i) = \sum_{m} Input(i+m)Kernel(m) \qquad (5.1)$$

2. Rectified Linear Unit (ReLU)

In the standard way of modeling, neuron's output in a neural network can be described with either tangent hyperbolic function (tanh) or sigmoid function (sigm). In terms of training time with gradient descend these saturating nonlinearity functions consume much more time when compared to non-saturating nonlinearity function, ReLU. This function can be replaced with previous functions used for increasing the nonlinear properties of the decision function without affecting the receptive fields of the convolution layer significantly. Usage of ReLU is also helpful to alleviate the vanishing gradient problem, which is the issue where the lower layers of the network train very slowly because the gradient decreases exponentially through the layers [46]. ReLU function is described in eq (5.2).

$$f_{\mathrm{ReLU}}(x) = \max(0, x) \qquad (5.2)$$

3. Pooling Layer

Pooling is an important concept used in CNNs. Although classification can be achieved without implementing any pooling, this process is commonly used in CNNs. The pooling layer is useful in reducing the number of parameters and amount of computation in the network. Briefly a pooling layer summarizes the outputs of neighboring groups of neurons in the same kernel map. This process is usually done by using one of the several non-linear functions. Max pooling function which is the most common function used in pooling, is chosen in this study.

4. Dropout Technique

Deep neural networks are very strong classification tools though those architectures, especially the ones consisting of large number of parameters, suffer from a serious problem called as overfitting. Overfitting describes an incorrect optimization problem for an artificial intelligence model, where the weights are too closely trained for a set of data, and this may result in false positive characterization. Combining the predictions of many different neural nets is a very successful way to handle this problem but this solution could become very expensive in terms of computational effort. The technique called as "dropout" is proposed to deal with this problem by combining different models in a very efficient way which only costs about a factor of two during training.

The main idea in this technique is randomly dropping neurons from the neural network during training stage with a probability of 0.5. Dropped neurons do not participate in the forward and back-propagation stages. So, every time an input is processed, the neural network basically samples a different architecture, but all architectures share weights. In the test stage, all neurons are used but their outputs multiplied by 0.5, which is a reasonable approximation of the geometric mean of the predictive distributions produced by dropout technique. In this way, expected value of an output neuron can be in the same range as in the training stages [47].

5. Fully Connected Layer

Neurons in a fully connected layer have connections to all activations in the previous layer, as seen in regular neural networks. Their activations can hence be computed with a matrix multiplication followed by a bias offset.

B. Implementation of CNN in Composite System Reliability Evaluation

In this section, implementation of the proposed approach is presented. The main motivation in this approach is reducing computation time of traditional MCS for evaluation of reliability indices.

In a typical MCS, reliability indices of a power system are computed in three main stages. First, states of power system components are sampled based on their probability distribution and then characterization of power system status is made using those samples. Finally, calculation of desired reliability indices is done based on the previously characterized sample series. Characterization of power systems is usually done by a power flow analysis. This process can be considered as a linear or nonlinear programming problem depending on power flow model chosen for an application. Performing a power flow analysis for every sample obtained by MCS can create a significant computational burden especially in large composite power systems, especially with high reliability. Systems with high reliability will need a significant sample size for converging with high accuracy.

The proposed method aims to reduce the amount of power flow analysis by using CNN as a pre-classifier. To achieve this a CNN is designed to characterize overall system status as success or failure by using the information gathered from sampled states of system components as an input. Then afterwards, power flow analysis is applied only for samples classified as failure states.

In this study, CNN architecture is implemented to classify system status obtained from both DC and AC power flow analysis. In addition, an algorithm is proposed for generating the required training samples for AI based classifiers used in power system reliability analysis. The proposed algorithm can create much more detailed datasets in a considerably shorter time. Non-sequential MCS approach is used to analyze performance of the proposed method because of the simplicity of model. In the following, proposed method is explained detailly in three subsections.

1. Creating Balanced Datasets

Creating a proper training dataset can be considered among one of the most important aspects that affects classification performance of an AI based classifiers. This subsection first describes the criteria used for determining training dataset, then highlights a common problem faced in classifying rare events called class imbalance problem and finally proposes a possible solution to this problem.

a. Size Issue in Training Datasets

The first step of creating a training dataset is to decide its size. Optimal size of a training dataset for an ANN commonly depends on parameters like input data, number of classes or number of neurons in the network and varies a lot based on the application. In power system reliability evaluation, AI based classifiers are generally used to increase the time efficiency of MCS. For this reason, the size of training dataset is determined proportionally to the sample size used in MCS.

The most common parameter to determine the number of samples used in MCS is coefficient of variation (COV). Typically, an acceptable value of the estimate of uncertainty is determined as the stopping criteria before the simulation and MCS run until the stopping criteria satisfied.

b. Class Imbalance Problem

One of the most common challenges faced in creating training dataset is called the class imbalance problem. This problem usually occurs when one or more target classes in a dataset are underrepresented (minor classes) in comparison with the other classes (major classes). Previous studies on this problem show that the negative effect of this problem in classification accuracy can be significantly reduced by applying down-sampling to the major class as well as applying an over-sampling to the minor class [48].

In a typical power system, occurrences of failure events (minor class) are much lower when compared to success events (major class). This feature of power systems creates a class imbalance problem and prevents effective training of AI based classifiers. The majority of the previous studies using AI based classifiers for characterization of power system states were using unproportioned training datasets by simply reducing the amount of the success states.

c. Proposed Solution

In this subsection, an algorithm is presented to deal with the class imbalance problem in power system state classification. This algorithm creates a training dataset which includes failure and success states in an equal proportion by applying down-sampling to major class (success) and over-sampling to the minor class (failure). It can be applied to any power system and AI based classifier without adding a considerable cost. The proposed algorithm consists of three main steps which are described followingly.

First, the size of the training dataset is determined. A looser stopping criterion is used in terms of COV for this purpose (e.g. 10%). Normally, COV calculation requires a system characterization for each sample obtained. In this step, the approximation described in equation (5.3) is used for this process to avoid additional computational burden of power flow analysis.

$$G_{total} \geq D_{total} \begin{cases} success, if\ true \\ failure,\ \ else \end{cases} \tag{5.3}$$

Where G_{total} describes sum of all generation in the system and D_{total} is for load. When COV reaches the previously described stopping criteria the total number of obtained samples is taken as the size of training dataset.

In the next step, power flow analysis is performed for a small portion of determined dataset size (e.g. 5%). The results obtained in this step are used for calculating the proportion of classes. If the number of success states obtained in this process is higher than number of failure states unavailability rate of all components is increased by multiplying a step size Δw. The process is repeated until an equal amount of success and failure states obtained by considering a tolerance.

Finally, a training dataset is created in size calculated in the first step by using unavailability values obtained in the previous step. A flowchart of this algorithm is provided in figure 5.1.

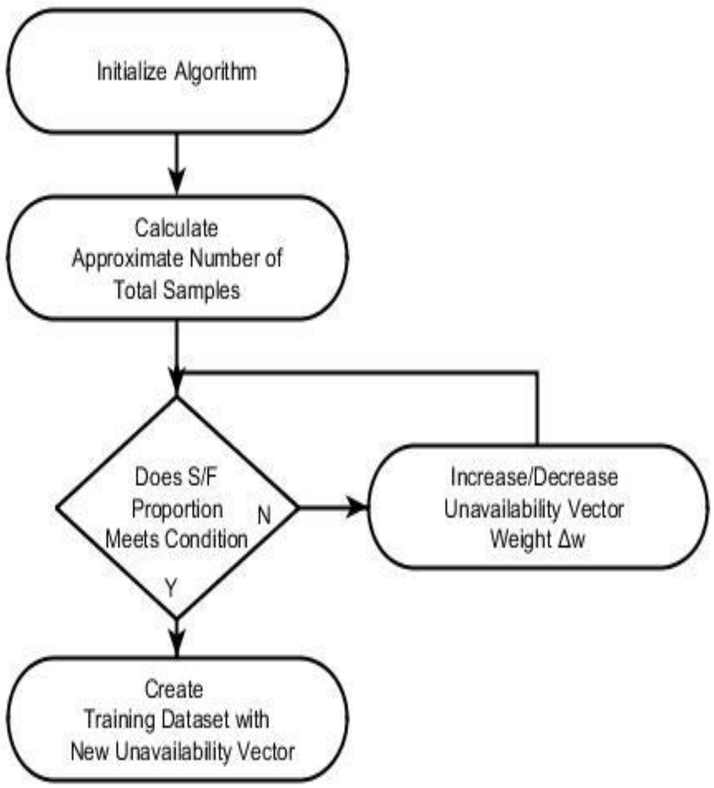

Figure 5.1: Proposed Algorithm used to create a Proportional Training Dataset

2. CNN Architectures for the Proposed Method

Most of the current methods use DC power flow, which is often called OPF, to evaluate the reliability of composite systems. OPF consists of a series of approximations in the usual power flow equations which reduces the problem down to a set of linear equations that are normally represented by nonlinear equations. OPF can find the optimum solution for a power system state significantly faster when compared to AC flow model, this feature makes it very suitable for power system reliability analysis. However, OPF ignores the effects of the voltage and reactive power constraints on the reliability indices. For this reason, reliability analysis performed by using OPF can be considered optimistic and can be different when compared to AC flow analysis. In other words, some states recognized as success states by using the OPF, may be characterized as failure by AC model since the failures of those states is usually caused by voltage and reactive power limit violations. For this reason, AC flow model is also considered in state characterization of obtained samples in this study. Extensive theoretical explanations and comparison for both models can be found in [49,50]. Followingly in this subsection, details of designed CNN architecture are explained for classifying DC and AC power flow models respectively.

a. Designed CNN Architecture

In this subsection the designed CNN architecture is described. It is essential to form the inputs of neural network by considering required information to be able to classify patterns. In this study, generation and demand information is used to create inputs. Since DC flow model considers only active power, one channel input is created with generation and demand information of a sampled state. As for the AC flow model, reactive power information is included to input of classifier as an additional channel.

The input of CNN classifier is created by using the information of maximum available generation and demand at each bus of composite power system. Input data described by equation (5.4).

$$Input_i = G_i - D_i \tag{5.4}$$

Where G represents maximum available generation and D is demand at bus i. The second channel used for classifying the states of AC flow model contains the information of maximum and minimum reactive power generation can be produced from each bus of power system as well as the demands for that bus. Input data of second channel is described in equation (5.5).

$$Input_i = Qmax_i - Qmin_i - D_i \tag{5.5}$$

Where Q_{max} and Q_{min} represents maximum and minimum limits of reactive generation, and D is demand at bus i. After the information is prepared a zero-center normalization is applied to create input vectors.

The designed architecture consists of three convolutional layers, one pooling layer and two fully connected layers. In the first convolution layer input is extracted to low level feature maps by using 24 kernels with size of 6x1. Then second and third layers of convolution are applied to gather more detailed features 48 and 96 kernels are applied with size of 4x1 and 2x1 respectively. Stride of 2x1 is applied to these convolutional layers. Following convolutional layers, a pooling layer is applied to the obtained extracted features. The output is applied to two fully connected layers. Each of these layers includes 24 neurons. Two layers of dropout are used with proportion of 0.5 for each of those fully connected layers. At the end a softmax function is used to accomplish binary classification. General diagram of designed CNN architectures is presented in figure 5.2 for a clearer understanding.

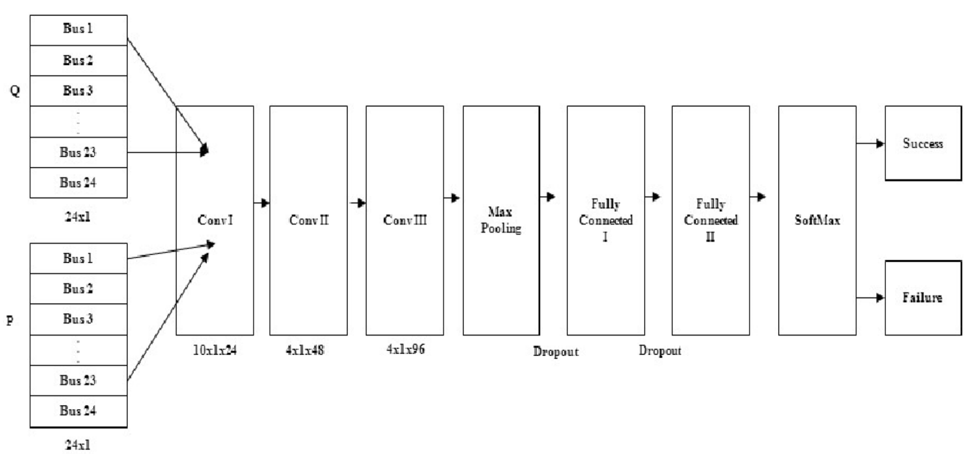

Figure 5.2: Overall Diagram of Proposed CNN Structure

3. Testing Stage

After proper training patterns have been obtained, CNN is trained and then used for characterizing overall system status of randomly generated samples within MCS. This process continues until MCS meets a previously determined stopping criterion which is COV for this study. When MCS reaches stopping criteria, power flow equations are applied only for states classified as failure. In this way, computational time necessary for evaluating the system reliability indices is reduced significantly. At the end of this procedure obtained results are compared with a Crude Monte Carlo Simulation (CMCS) benchmark. The performance of the proposed method is evaluated by the metrics described in Chapter 2.

Application procedure of the proposed deep CNN and MCS combination is described in steps above. For better understanding, a flowchart for the proposed method is given in figure 5.3.

Performance of proposed method for both AC and DC flow models is shown on two different case studies for constant and varying load levels respectively. Case studies are described in details followingly.

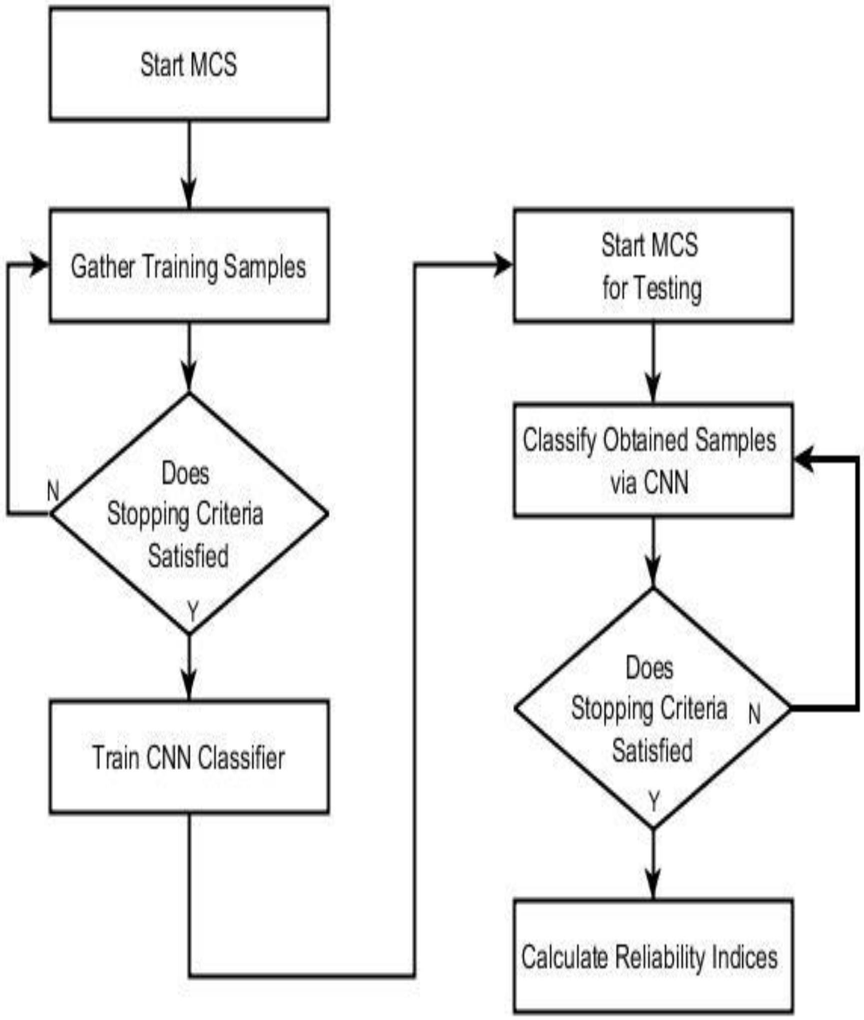

Figure 5.3: General Structure of Proposed Method

C. Case Studies and Results

In this section, two case studies were conducted on IEEE RTS to show the performance of the proposed method. In the first case study the system load is considered as constant at its annual peak value while in the second case study varying hourly load data of RTS is used. Both case studies are implemented for DC and AC flow model. It should to be mentioned that transmission lines in this study are considered available at all the time since failure rate of transmission lines are much lower than generation units. The capacity constraints of transmission lines are, however, considered. All the simulations are performed using MATLAB (2017b) platform on a PC with Intel Core i7-4510 CPU (~2.6GHz), 16 GB Memory.

Simulation results for case studies are discussed in following subsections.

1. Constant Load Level

In this case, the system is tested on single area IEEE RTS for single load level described as 2850 MW (annual peak). There are 10 generation buses in RTS which are considered as input vector. To train the classifier, initial COV for training dataset creator algorithm is chosen as 10%. After obtaining adequate number of samples, the training of proposed CNN is completed within 100 iterations. After classifier is successfully trained, the proposed system is tested until COV reaches the limit of $\leq 1\%$ as stopping criteria for both AC and DC flow equation model. After testing is completed, 109743 samples are classified by DC-OPF with 100470 successes and 9273 failure states. Similarly, 53194 samples are classified by AC-OPF with 44643 successes and 8551 failure states.

Table 5.1 presents the comparative results between CMCS (Crude Monte Carlo Simulation) and the proposed approach for DC flow model while Table 5.2 provides results of similar analysis for AC flow model. Performance indices are calculated as described in Chapter 2.

Table 5.1: Performance Comparison of CNN-MCS & CMCS for Constant Load

(DC Flow Equation Model)

	CMCS	**CNN**
Success States	100470	100474
Failure States	9273	9269
Loss of Load Probability	0.0845	0.0845
Sensitivity	N/A	0.999
Specificity	N/A	0.999
Analysis Time (Sec)	5321	182

Table 5.2: Performance Comparison of CNN-MCS & CMCS for Constant Load

(AC Flow Equation Model)

	CMCS	**CNN**
Success States	44644	44648
Failure States	8550	8546
Loss of Load Probability	0.1607	0.1607
Sensitivity	N/A	0.999
Specificity	N/A	0.999
Analysis Time (Sec)	139638	112

It is clear in Table 5.1 that the proposed method shows an outstanding performance for both in classification accuracy and reliability evaluation with significantly reduced computational effort for DC flow model.

Table 5.2 shows the classification performance of the proposed method. Proposed CNN classifier can characterize the system states with a small error rate in terms of both classification accuracy and reliability evaluation. It should be noted that computational time for AC power flow model is significantly reduced compared to reduction in DC flow model since this model requires nonlinear programming to solve AC flow equations. It should be noted that the time for AC analysis is approximately 39 hours in CMCS where as in the CNN approach it is less than two minutes.

2. Varying Hourly Load Level

In this case, the system is tested on single area IEEE RTS for original annual hourly load data. All 24 buses of RTS are considered as input vector as described in (eq. 5.4 - 5.5). To train the classifier, initial COV for training dataset creator algorithm is chosen as 10%. After obtaining adequate number of samples, the training of proposed CNN is completed within 200 iteration. After classifier is successfully trained, the proposed system is tested until COV reaches the limit of $\leq 1\%$ as stopping criteria for both AC and DC flow equation model. After testing is completed, 7466186 samples are classified by DC-OPF with 7457892 successes and 8294 failure states. Similarly, 5701891 samples are classified by AC-OPF with 5692084 successes and 9807 failure states.

Table 5.3 presents the comparative results between CMCS and the proposed approach for DC flow model while Table 5.4 provides results of similar analysis for AC flow model. It is clear in Table 5.3 that the proposed method shows very good accuracy in classification of system states and predicts reliability indices with a small error rate. Table 5.4 also shows that proposed method can significantly reduce the computational effort for AC flow model.

Table 5.3: Performance Comparison of CNN-MCS & CMCS in Varying Load

(DC Flow Equation Model)

	CMCS	**CNN**
Success States	7457892	7457973
Failure States	8294	8213
Loss of Load Probability	0.0011	0.0011
Sensitivity	N/A	0.999
Specificity	N/A	0.999
Analysis Time (Sec)	354861	10516

Table 5.4 shows that proposed method has an outstanding performance for both in classification accuracy and reliability evaluation with significantly reduced computational effort in AC flow model.

Table 5.4: Performance Comparison of CNN-MCS & CMCS in Varying Load

(AC Flow Equation Model)

	CMCS	**CNN**
Success States	5692084	5692198
Failure States	9807	9693
Loss of Load Probability	0.0017	0.0017
Sensitivity	N/A	0.999
Specificity	N/A	0.999
Analysis Time (Sec)	16842874	9147

D. Conclusion

In this study, MCS is combined with a deep CNN structure to evaluate composite system reliability to increase the computational efficiency of simulation.

It is critical for any binary classification problem to create a balanced dataset. In this paper, a new algorithm is proposed to create a proper dataset by applying down-sampling to success states as well as applying over-sampling to failure states. The proposed algorithm can be applicable for any artificial intelligence-based method without any additional cost.

The proposed method is tested for both DC and AC flow models by applying one and two channel CNN structure respectively. Generation and demand information of a sampled state is utilized to create input vector of classifier.

Two case studies are conducted to demonstrate the performance of the proposed MCS-CNN combination based on single level and hourly load model respectively. Sensitivity and Specificity parameters are used to show classification performance of CNN classifier while LOLP chosen as metrics to demonstrate reliability evaluation of proposed method.

Obtained results on fixed peak load (case 1) and varying original hourly load (case 2) show that the proposed method can accurately characterize system states for both DC and AC flow model and therefore reliability indices of RTS can be evaluated with a negligible error rate in a significantly less simulation time. It can be seen that AC and DC models can give quite different results on the same test system.

Computational efficiency for classifying of AC flow model is much higher than DC flow model since AC flow equations requires nonlinear programming techniques while DC flow equations can be solved with linear techniques. The results are obtained with high accuracy with COV equal or less than .01. The computation times depend on the COV used.

CHAPTER VI

CONCLUDING REMARKS

In this chapter, conclusions are made to summarize the work reported in this dissertation. Also, some suggestions for future investigations in this research arena are given.

A. Conclusion

In this dissertation, a novel method to evaluate reliability indices for composite power systems is introduced with combination of MLL and MCS. The proposed method is implemented for MLKNN and MLRBF classifiers to identify status of buses. The case studies show that the method significantly reduces the computational burden of MCS. Additionally, this method advances the state of the art of using machine learning in power system reliability evaluation from the previous methods by including computation of bus indices and the transmission line failures.

Moreover, the work has been done to show that the proposed method can be combined with well-known variance reduction technique IS. The outcomes for this approach show this methodology improves time efficiency of MCS even further.

Finally, deep learning structure is investigated to evaluate composite system reliability evaluation through MCS. CNN, well known deep learning topology, is implemented to characterize sampled system states for both AC and DC flow model. The results show that computational efficiency for classifying of AC flow model is much higher than DC flow model since AC flow equations requires nonlinear programming techniques while DC flow equations can be solved with linear techniques. The results obtained prove that the proposed architecture performs state characterization with a high accuracy with COV equal or less than .01.

The importance of reducing the computational time can be understood by two examples. In Monte Carlo Simulation, the accuracy of convergence is very important. Convergence is measured by the COV, smaller COV means better convergence. Now the sample size (or computational time) is inversely proportional to COV or directly proportional to accuracy of convergence. The proposed method reduces the required time for reliability analysis considerably for the same level of accuracy defined by the coefficient of variation. Alternatively, for the same time this allows convergence to a higher level of accuracy. Another example is that for optimal planning of resources, reliability studies may need to be done many times [51]. So, the reduction of computational time helps in optimal planning.

This study demonstrates that the application of the proposed methods on composite power system reliability evaluation accurately determines the states status with a huge speed up compared with OPF based Monte Carlo Simulation methods. In short, the work reported here contributes in the following ways;

1. Enhance the computational efficiency and accuracy using machine learning and variance reduction techniques.
2. Enhance the scope by including computation of bus indices in addition to the system indices while using machine learning.
3. Include the AC power flow and show how it can make a difference in the computed reliability.

B. Future Work and Suggestions

Performance of power system reliability assessment techniques can be significantly enhanced in conjunction with machine learning methods. Even though many important achievements have been made in this area, more research is yet to be done. In this final section, the following aspects relating to future research are outlined below;

- Deep multi-label learning topologies can be investigated to increase the performance of MCS even further. The use of parallel computation in conjugation with deep learning algorithms should also be considered as an important boost factor of computational efficiency.
- Renewable energy sources are being utilized with an increasing penetration and the power plants including renewable sources are built and integrated into existing power systems for various reasons. The AI based computational methods can be used to identify the correlation between time dependent structure of renewable energy and system load for efficient computation of reliability indices for systems integrated with renewable energy.
- Failures caused by transmission line capacity limitations are successfully classified within the scope of this dissertation, however the implementation of proposed method is only performed for DC flow model. More studies can be done to perform classification of these failures for AC flow model.
- Power system reliability assessment with Electrical Vehicle integration is an important research topic for future generation power systems. Implementation of

machine learning techniques for both classification and load estimation is a field that requires more attention.

- So far, the machine learning methods in reliability analysis have focused on the planning stage. With the encouraging results in computational efficiency, these methods can be investigated for operational planning.

REFERENCES

[1] Singh, Chanan, and Roy Billinton. *System reliability, modelling and evaluation*. Vol. 769. London: Hutchinson, 1977.

[2] Singh, C., and P. Jirutitijaroen. "Monte Carlo simulation techniques for transmission systems reliability analysis." *A Tutorial Paper presented at IEEE Power Engineering Society General meeting, Tampa, Florida*. 2007.

[3] Zhaohong, B., & Xifan, W. (2002). Studies on variance reduction technique of Monte Carlo simulation in composite system reliability evaluation. *Electric Power Systems Research*, *63*(1), 59-64.

[4] Singh, C., & Mitra, J. (1997). Composite system reliability evaluation using state space pruning. *IEEE Transactions on Power Systems*, *12*(1), 471-479.

[5] Saraiva, J. T., Miranda, V., & Pinto, L. M. V. G. (1995, May). Generation/transmission power system reliability evaluation by Monte Carlo simulation assuming a fuzzy load description. In *Power Industry Computer Application Conference, 1995. Conference Proceedings. 1995 IEEE* (pp. 554-559). IEEE.

[6] Miranda, V., de Magalhães Carvalho, L., Da Rosa, M. A., Da Silva, A. M. L., & Singh, C. (2009). Improving power system reliability calculation efficiency with EPSO variants. *IEEE Transactions on Power Systems*, *24*(4), 1772-1779.

[7] Samaan, N., & Singh, C. (2002). Adequacy assessment of power system generation using a modified simple genetic algorithm. *IEEE Transactions on Power Systems*, *17*(4), 974-981.

[8] Samaan, N., & Singh, C. (2003, July). Assessment of the annual frequency and duration indices in composite system reliability using genetic algorithms. In *Power Engineering Society General Meeting, 2003, IEEE* (Vol. 2, pp. 692-697). IEEE.

[9] Earla, R., Mitra, J., & Patra, S. B. (2004, August). A particle swarm based method for composite system reliability analysis. In *North American Power Symposium*.

[10] Wang, L., & Singh, C. (2008). Population-based intelligent search in reliability evaluation of generation systems with wind power penetration. *IEEE Transactions on Power Systems*, *23*(3), 1336-1345.

[11] Singh, C., & Wang, L. (2008). Role of artificial intelligence in the reliability evaluation of electric power systems. *Turkish Journal of Electrical Engineering & Computer Sciences*, *16*(3), 189-200.

[12] da Silva, A. M. L., de Resende, L. C., da Fonseca Manso, L. A., & Miranda, V. (2007). Composite reliability assessment based on Monte Carlo simulation and artificial neural networks. *IEEE Transactions on Power Systems, 22*(3), 1202-1209.

[13] Pindoriya, N. M., Jirutitijaroen, P., Srinivasan, D., & Singh, C. (2011). Composite reliability evaluation using Monte Carlo simulation and least squares support vector classifier. *IEEE Transactions on Power Systems*, *26*(4), 2483-2490.

[14] M. V. F. Pereira and N. J. Balu, "Composite generation/transmission reliability evaluation," Proc. IEEE, vol. 80, no. 4, pp. 470–491, Apr. 1992.

[15] W. Li and R. Billinton, "Effect of bus load uncertainty and correlation in composite system adequacy evaluation," IEEE Trans. Power Syst., vol. 6, no. 4, pp. 1522–1529, Nov. 1991

[16]] A. C. G. Melo, M. V. F. Pereira, and A. M. L. da Silva, "A conditional probability approach to the calculation of frequency and duration indices in composite reliability evaluation," IEEE Trans. Power Syst., vol. 8, no. 3, pp. 1118–1125, Aug. 1993.

[17] F. F. C. Véliz, C. L. T. Borges, and A. M. Rei, "A comparison of load models for composite reliability evaluation by nonsequential Monte Carlo simulation," IEEE Trans. Power Syst., vol. 25, no. 2, pp. 649–656, May 2010

[18] J. Mitra and C. Singh, "Pruning and simulation for determination of frequency and duration indices of composite power systems," IEEE Trans. Power Syst., vol. 14, no. 3, pp. 899–905, Aug. 1999.

[19]] R. Billinton and W. R. Li, Assessment of Electric Power Systems Using Monte Carlo Methods. New York, NY, USA: Plenum, 1994

[20]] H. Kim and C. Singh, "Reliability modeling and simulation in power systems with aging characteristics," IEEE Trans. Power Syst., vol. 25, no. 1, pp. 21–28, Feb. 2010.

[21] P. Wang and R. Billinton, "Time sequential distribution system reliability worth analysis considering time varying load and cost models," IEEE Trans. Power Del., vol. 14, no. 3, pp. 1046–1051, Jul. 1999.

[22] Sankarakrishnan, A., and Roy Billinton. "Sequential Monte Carlo simulation for composite power system reliability analysis with time varying loads." *IEEE Transactions on Power Systems* 10.3 (1995): 1540-1545.

[23] Billinton, R., et al. "Reliability assessment of composite generation and transmission systems." *IEEE Tutorial, IEEE Winter Power Meeting*. 1990

[24] Benidris, Mohammed, and Joydeep Mitra. "Reliability and sensitivity analysis of composite power systems considering voltage and reactive power constraints." *IET Generation, Transmission & Distribution* 9.12 (2015): 1245-1253.

[25] Luo, X., A. D. Patton, and C. Singh. "Real power transfer capability calculations using multi-layer feed-forward neural networks." *IEEE Transactions on Power Systems* 15.2 (2000): 903-908.

[26] Force, RTS Task. "IEEE reliability test system." *IEEE Trans. on PAS* 98.6 (1979): 2047-2054.

[27] Tsoumakas, G., & Katakis, I. (2006). Multi-label classification: An overview. *International Journal of Data Warehousing and Mining*, *3*.

[28] Zhang, M. L., & Zhou, Z. H. (2007). ML-KNN: A lazy learning approach to multi-label learning. *Pattern Recognition*, *40*(7), 2038-2048.

[29] Zhang, M. L., & Zhou, Z. H. (2006). Multilabel neural networks with applications to functional genomics and text categorization. *IEEE Transactions on Knowledge and Data Engineering*, *18*(10), 1338-1351.

[30] Thabtah, F. A., Cowling, P., & Peng, Y. (2004, November). MMAC: A new multi-class, multi-label associative classification approach. In *Data Mining, 2004. ICDM'04. Fourth IEEE International Conference on* (pp. 217-224). IEEE.

[31] Wu, T. F., Lin, C. J., & Weng, R. C. (2004). Probability estimates for multi-class classification by pairwise coupling. *Journal of Machine Learning Research*, *5*(Aug), 975-1005.

[32] Bishop, Christopher M. Neural networks for pattern recognition. Oxford University Press, 1995.

[33] Zhang, Min-Ling. "MLRBF Neural Networks for Multi-Label Learning." Neural Processing Letters 29.2 (2009): 61-74.

[34] da Silva, Armando M. Leite, Reinaldo AG Fernandez, and Chanan Singh. "Generating capacity reliability evaluation based on Monte Carlo simulation and cross-entropy methods." *IEEE Transactions on Power Systems* 25.1 (2010): 129-137.

[35] González-Fernández, Reinaldo A., et al. "Composite systems reliability evaluation based on Monte Carlo simulation and cross-entropy methods." *IEEE Transactions on Power Systems* 28.4 (2013): 4598-4606.

[36] Rubinstein, Reuven Y., and Dirk P. Kroese. *Simulation and the Monte Carlo method*. Vol. 10. John Wiley & Sons, 2016.

[37] T. Homem-deMello and R.Y Rubinstein, Estimation of Rare Event Probabilities Using Cross-Entropy, in Proc. Winter Simulation Conf., San Diego, CA, USA, Dec 2002, vol. 1, pp. 310-319.

[38] Luo, Xiaochuan, Chanan Singh, and Alton D. Patton. "Power system reliability evaluation using learning vector quantization and Monte Carlo simulation." *Electric Power Systems Research*, 66.2 (2003): 163-169.

[39] LeCun, Yann, Yoshua Bengio, and Geoffrey Hinton. "Deep learning." *Nature* 521.7553 (2015): 436.

[40] Deng, Li, and Dong Yu. "Deep learning: methods and applications." *Foundations and Trends® in Signal Processing*7.3–4 (2014): 197-387.

[41] Hubel, David H., and Torsten N. Wiesel. "Receptive fields, binocular interaction and functional architecture in the cat's visual cortex." *The Journal of Physiology* 160.1 (1962): 106-154.

[42] Felleman, Daniel J., and DC Essen Van. "Distributed hierarchical processing in the primate cerebral cortex." *Cerebral Cortex (New York, NY: 1991)* 1.1 (1991): 1-47.

[43] Taigman, Y., Yang, M., Ranzato, M. & Wolf, L. Deepface: closing the gap to human-level performance in face verification. In *Proc. Conference on Computer Vision and Pattern Recognition* 1701–1708 (2014).

[44] Goodfellow, Ian, et al. *Deep learning*. Vol. 1. Cambridge: MIT press, 2016.

[45] Hu, Wei, et al. "Deep convolutional neural networks for hyperspectral image classification." *Journal of Sensors* 2015 (2015).

[46] Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E. Hinton. "Imagenet classification with deep convolutional neural networks." *Advances in Neural Information Processing Systems*. 2012.

[47] Srivastava, Nitish, et al. "Dropout: a simple way to prevent neural networks from overfitting." *The Journal of Machine Learning Research* 15.1 (2014): 1929-1958.

[48] Chawla, Nitesh V., et al. "SMOTE: synthetic minority over-sampling technique." *Journal of Artificial Intelligence Research* 16 (2002): 321-357.

[49] Wang, Peng, et al. "Reliability assessment of power systems considering reactive power sources." *Power & Energy Society General Meeting, 2009. PES'09. IEEE*. IEEE, 2009.

[50] Overbye, Thomas J., Xu Cheng, a Yan Sun. "A comparison of the AC and DC power flow models for LMP calculations." *System Sciences, 2004. Proceedings of the 37th Annual Hawaii International Conference on*. IEEE, 2004.

[51] Singh, C., Jirutitijaroen, P. and Mitra, J., *Electric power grid reliability evaluation: models and methods*, 2019, Wiley-IEEE Press.