

PROCESS FAULT PREDICTION AND PROGNOSIS
BASED ON A HYBRID TECHNIQUE

by Galagedarage Don Mihiran Pathmika

a Thesis submitted to the School of Graduate Studies in partial fulfillment of the
requirements for the degree of

Master's in Engineering

Memorial University of Newfoundland

May 2019

St. John's, Newfoundland and Labrador

ABSTRACT

The present study introduces a novel hybrid methodology for fault detection and diagnosis (FDD) and fault prediction and prognosis (FPP). The hybrid methodology combines both data-driven and process knowledge driven techniques. The Hidden Markov Model (HMM) and the auxiliary codes detect and predict the abnormalities based on process history while the Bayesian Network (BN) diagnoses the root cause of the fault based on process knowledge. In the first step, the system performance is evaluated for fault detection and diagnosis and in the second step, prediction and prognosis are evaluated. In both cases, an HMM trained with Normal Operating Condition data is used to determine the log-likelihoods (LL) of each process history data string. It is then used to develop the Conditional Probability Tables of BN while the structure of BN is developed based on process knowledge. Abnormal behaviour of the system is identified through HMM. The time of detection of an abnormality, respective LL value, and the probabilities of being in the process condition at the time of detection are used to generate the likelihood evidence to BN. The updated BN is then used to diagnose the root cause by considering the respective changes of the probabilities. Performance of the new technique is validated with published data of Tennessee Eastman Process. Eight of the ten selected faults were successfully detected and diagnosed. The same set of faults were predicted and prognosed accurately at different levels of maximum added noise.

Keywords: *HMM, Bayesian Network, Fault Prediction, Prognosis, Fault Diagnosis*

To my loving parents, G.D. Somapala and K. Indrani,
and
my loving wife, Lekshika.

ACKNOWLEDGMENT

My foremost sincere gratitude is expressed to my supervisor, Prof. Faisal Khan, who gave me the opportunity to carry out this research work and for the immense help and guidance given throughout the project work.

Also, I thankfully acknowledge the financial support provided by the Natural Science and Engineering Council of Canada and the Canada Research Chair (CRC) Tier I Program in offshore safety and risk engineering.

I would also like to thank Prof. George Mann, Mr. Samith Rathnayake, Dr. Thumeera Wanasinghe, Dr. Awantha DewageDon, and Mr. Dilhan Balage for all the support they gave me to make this whole process a success.

In addition to that, I would like to express my thanks to all the academic staff members of the Faculty of Engineering and Applied Science, for their assistance throughout my stay.

In conclusion, I would like to express my pardon if I have inadvertently omitted the name of those to whom thanks is due.

Mihiran, Galagedarage Don

TABLE OF CONTENTS

Abstract	ii
Acknowledgment	iv
Table of Contents	v
List of tables	ix
List of figures	x
List of Abbreviations and symbols	xii
list of appendices	xiv
Chapter 1 : Introduction and overview	1
1.1 <i>Introduction</i>	1
1.2 <i>Objectives</i>	2
1.3 <i>Co-authorship Statement</i>	2
1.4 <i>Thesis outline</i>	3
Chapter 2 : Dynamic process fault detection and diagnosis based on a combined approach of hidden markov and bayesian network model	5
2.1 <i>Introduction</i>	6
2.2 <i>Literature Survey</i>	8
2.2.1 <i>History of Data-based FDD</i>	10
2.2.2 <i>Hidden Markov Model (HMM)</i>	13

2.2.3	<i>Mathematical Formulation</i>	15
2.2.4	<i>HMM-based FDD</i>	17
2.2.5	<i>HMM Tool Box for MATLAB</i>	24
2.2.6	<i>BN based FDD</i>	25
2.3	<i>Main observations from the Literature Survey</i>	26
2.4	<i>The Methodology</i>	28
2.4.1	<i>Illustration of the methodology using tank model</i>	29
2.4.2	<i>Training of HMM_1</i>	30
2.4.3	<i>Development of the Bayesian Network</i>	31
2.4.4	<i>Development of Qualitative BN</i>	32
2.4.5	<i>Mapping the SDG to BN</i>	32
2.4.6	<i>Training the BN</i>	34
2.4.7	<i>Introduction of Likelihood Evidence to the trained BN</i>	36
2.5	<i>Application of the methodology in Tennessee Eastman (TE) Process</i>	37
2.5.1	<i>Tennessee Eastman Process</i>	37
2.5.2	<i>Fault detection by HMM</i>	38
2.5.3	<i>Bayesian Network Training</i>	41
2.6	<i>Results and Discussion</i>	41
2.6.1	<i>Fault detection step</i>	41

2.6.2	<i>A method to simplify the Matlab coding task</i>	45
2.7	<i>Conclusions</i>	46
	<i>References</i>	47

Chapter 3 : process fault prediction and prognosis using a hybrid model.....58

3.1	<i>Introduction and Review of the Relevant Literature</i>	59
3.2	<i>Preliminaries</i>	76
3.2.1	<i>Hidden Markov Model (HMM)</i>	76
3.2.2	<i>Bayesian Networks (BNs)</i>	79
3.3	<i>The Methodology</i>	80
3.3.1	<i>Data preprocessing</i>	81
3.3.2	<i>Training of HMM and prediction of the nth data string</i>	82
3.3.3	<i>Development of structure and training of BN</i>	89
3.3.4	<i>Introduction of Likelihood Evidence to the trained BN</i>	90
3.3.5	<i>Prognosis using BN</i>	90
3.4	<i>Results and Discussion</i>	92
3.5	<i>Conclusions</i>	95
	<i>References</i>	96

Chapter 4 : Auxiliary codes for fault prognosis process of tennessee eastman process using a hybrid model (CPL1.0)101

4.1	<i>Motivation and Significance</i>	102
4.2	<i>Software Description</i>	103
4.3	<i>Software Architecture</i>	103
4.4	<i>Sample code snippets analysis.</i>	104
4.5	<i>Illustrative Example</i>	106
4.6	<i>Impact</i>	108
4.7	<i>Conclusions</i>	110
4.8	<i>References</i>	110
Chapter 5 : Summary and Conclusion		111
APPENDIX A: Detection of faults from fault A to J		x
APPENDIX B: Detection of faults from fault A to J		xiii

LIST OF TABLES

<i>Table 1: Discrete HMM elements (de Almeida & Park, 2008)</i>	16
<i>Table 2-2: Summary of HMM-based fault diagnosis</i>	24
<i>Table 2-3: FDD in process engineering applications</i>	27
<i>Table 4: Continuous process variables of TE chemical process (Amin, Imtiaz and Khan, 2018)</i>	39
<i>Table 5: Faults in the TE process and their respective root causes</i>	40
<i>Table 2-6: Detection times of All ten faults</i>	43
<i>Table 2-7: Diagnosis of fault A using BN</i>	43
<i>Table 8: Performance Comparison of the proposed technique</i>	44
<i>Table 2-9: Sequences for a node with two parent nodes</i>	46
<i>Table 3-1: Faults in the TE process and their respective causes</i>	82
<i>Table 3-2: Description of notations used</i>	85
<i>Table 3-3: Prognosis of fault A</i>	93
<i>Table 3-4: Summary of results</i>	94
<i>Table 4-1: Code Meta Data</i>	109

LIST OF FIGURES

<i>Figure 1-1: The contribution of each chapter to the formation of the thesis</i>	<i>4</i>
<i>Figure 2: First order HMM in the form of BN representation (de Almeida & Park, 2008)</i>	<i>16</i>
<i>Figure 3: Input-output relationship of HMM (de Almeida & Park, 2008).....</i>	<i>23</i>
<i>Figure 2-3: Flowchart of BN based Fault Diagnosis (Cai, Huang and Xie, 2017)</i>	<i>28</i>
<i>Figure 2-4: The overall procedure of FDD.....</i>	<i>30</i>
<i>Figure 2-5: Tank model</i>	<i>30</i>
<i>Figure 2-6: Procedure to develop the Quantitative BN.....</i>	<i>31</i>
<i>Figure 2-7: Development of SDG.....</i>	<i>33</i>
<i>Figure 2-8: Mapping of SDG to BN (Mallick, 2013).....</i>	<i>33</i>
<i>Figure 2-9: The BN representing the Tank Model.....</i>	<i>34</i>
<i>Figure 2-10: Different Zones in NOC Data.....</i>	<i>34</i>
<i>Figure 2-11: Output of HMM</i>	<i>37</i>
<i>Figure 2-12: Process Flow Diagram of the TE process (Ding, 2014)</i>	<i>40</i>
<i>Figure 2-13: BN developed for TE process</i>	<i>42</i>
<i>Figure 2-14: Detection of Fault A with HMM trained with NOC data.....</i>	<i>42</i>
<i>Figure 3-1: Different machine learning techniques (The Mathworks, 2016).....</i>	<i>60</i>
<i>Figure 3-2: Different techniques in Clustering</i>	<i>64</i>

<i>Figure 3-3: Establishing HMM and BN</i>	<i>81</i>
<i>Figure 3-4: Overview of prediction and prognosis</i>	<i>81</i>
<i>Figure 3-5: Training and application process of an HMM.....</i>	<i>84</i>
<i>Figure 3-6:3D matrix of data</i>	<i>84</i>
<i>Figure 3-7: Training Curve for HMM.....</i>	<i>88</i>
<i>Figure 3-8: Prediction of nth data string using trained HMM.....</i>	<i>88</i>
<i>Figure 3-9: Refining process of the initial prediction</i>	<i>89</i>
<i>Figure 3-10: Predicted Vs. Actual LL of Fault A</i>	<i>89</i>
<i>Figure 3-11: BN for the TE process (Pathmika and Khan, 2018a).....</i>	<i>91</i>
<i>Figure 3-12: Procedure for prognosis of the fault</i>	<i>92</i>
<i>Figure 3-13: Fault A prediction using HMM</i>	<i>93</i>
<i>Figure 4-1: The procedure to use CPL1.0.....</i>	<i>104</i>
<i>Figure 4-2: The input data format (Chapter 3, Figure 3-6).....</i>	<i>104</i>
<i>Figure 4-3: Safe and danger zones of each parameter reading (Chapter 2, Figure 2-8)</i>	<i>105</i>
<i>Figure 4-4: The training curve of HMM (Chapter 3, Figure 3-7).....</i>	<i>107</i>
<i>Figure 4-5: The actual LL variation of fault</i>	<i>107</i>
<i>Figure 4-6: Predicted LL variation of the fault with 10⁻² % of noise.....</i>	<i>108</i>

LIST OF ABBREVIATIONS AND SYMBOLS

AEM	Abnormal Event Management
AHMMASM	Adaptive HMM with Anomaly States Model
AI	Artificial Intelligence
ANFIS	Adaptive Neuro-Fuzzy Inference System
ANN	Artificial Neural Network
ARIMA	Auto-Regressive Integrated Moving Average
ARM	Auto-Regressive Model
BIC	Bayesian Information Criterion
BN	Bayesian Network
BW	Baum-Welch
CBM	Condition Based Maintenance
CP	Conditional Probabilities
CPT	Conditional Probability Tables
CSTR	Continuous Stirred Tank Reactor
CUSUM	Cumulative Sum
CVA	Canonical Variate Analysis
CVTA	Canonical Variable Trend Analysis
DHMM	Discrete Hidden Markov Model
DMFD	Dynamic Multiple Fault Diagnosis
DSS	Dependence Structure Subspace
DT	Decision Trees
EEMD	Empirical Mode Decomposition
EM	Expectation Maximization
EMD	Empirical Mode Decomposition
FDD	Fault Detection and Diagnosis
FFT	Fast Fourier Transform
FPP	Fault Prediction and Prognosis
GLAR	Generalized Linear Auto-Regression
GMM	Gaussian Mixture Model
GP	Genetic Programming
GPR	Gaussian Process Regression
HMM	Hidden Markov Model
ICA	Independent Component Analysis
ISF	Imperial Smelting Furnace
k-NN	K Nearest Neighbors
KPA	Kernel Principal Component Analysis

LIST OF ABBREVIATIONS AND SYMBOLS (CONTD...)

LAD	Logical Analysis of Data
LE	Likelihood Evidence
LL	Log Likelihood
LLNF	Locally Linear Neuro-Fuzzy
LRC	Logistic Regression Classifier
LSSVM	Least Squares Support Vector Machines
LSTM	Long-Short-Term Memory
MAV	Mean Absolute Value
MDS	Margin Distribution Subspace
MHMM	Modified Hidden Markov Model
MPLS	Multiway Partial Least Squares
NCA	Neighborhood Component Analysis
NLGBN	Nonlinear Gaussian Belief Network
NOC	Normal Operating Condition
OoBN	Object-Oriented Bayesian Networks
PCA	Principal Component Analysis
PF	Particle Filter
PFD	Process Flow Diagram
PP	Prior Probabilities
PSO	Particle Swarm Optimization
QDA	Quadratic Discriminant Analysis
QTA	Qualitative Trend Analysis
RDF	Random Decision Forests
RF	Random Forests
RTKBS	Real-Time Knowledge-Based System
RUL	Remaining Useful Life
SDG	Sign Directed Graphs
SMART	Self-Monitoring and Reporting Technology
SOM	Self -Organizing Map
SRCE	Swarm Rapid Centroid Estimation
SVD	Singular Value Decomposition
SVMR	Support Vector Machine Regression
TE	Tennessee Eastman
WLS	Wavelet Lifting Scheme
WT	Wavelet Transform

LIST OF APPENDICES

<i>Appendix A: Detection of faults B to J</i>	<i>x</i>
<i>Appendix B: Prediction of faults B to J</i>	<i>xiii</i>

CHAPTER 1 : INTRODUCTION AND OVERVIEW

1.1 Introduction

This thesis presents validated techniques for both early ‘detection and diagnosis’ and ‘prediction and prognosis’. The proposed technique addresses the problem of early detection and diagnosis of industrial process faults.

Hybrid approaches have shown improved performance in fault detection and diagnosis (FDD) over individual techniques (Venkat *et al.*, 2003). The current study proposes a hybrid technique which consists of a component which considers the process historical data and another component which extracts process knowledge for this FDD process. As a further development, the system is upgraded such that it can be used to predict potential faults and prognose the most probable root causes (FPP). The core of this research is the development of a methodology to combine the two systems, data-driven and knowledge-driven, to perform the tasks FDD and FPP. The Hidden Markov model (HMM) is used as the data-driven technique while Bayesian Networks (BN) is used as the knowledge-driven technique which acquires the process knowledge to FDD and FPP processes. The prediction and its weight giving out from HMM are used to establish BN. The diagnosis and prognosis are done based on the developed BN. The proposed techniques are tested for performance by applying in a real-world problem called FDD in Tennessee Eastman process (TE). The proposed hybrid technique can early detect and predict all the selected faults while successfully diagnose and prognose 80% of the faults.

CPL1.0 is the software code developed to determine the Conditional Probability Tables (CPT), Prior Probabilities (PP), and Likelihood Evidences (LE). It is presented as an

integral component of the study and made available online for future users. There are no such techniques published in the literature to satisfy similar needs arise in future studies. The entire process proposed is highly transparent and can be reproduced. The code is developed to make the modifications more convenient. Future researchers working on the BN combined with another machine learning techniques can use the developed code with minimum alterations.

1.2 Objectives

The primary objective is to develop a novel robust methodology to detect and diagnose process faults from a safety perspective. Secondly, to develop an extended methodology such that it can predict and prognose the faults with added noise. Thirdly, to present a complete software code which can be utilized in future developments related to hybrid machine learning techniques where BN is a component.

1.3 Co-authorship Statement

Galagedarage Don Mihiran Pathmika is the principal author of this thesis. He has undertaken the research and prepared the first draft. Professor Faisal Khan, the co-author of this thesis, shared the problem and conceptualized the methodology. Prof Khan guided author throughout the entire process of the methodology development, testing, validations and its application development. Further, the co-author contributed by reviewing, and revising the thesis. The software code and analysis of results were solely contributed by the principal author, and the results were validated for correctness by the co-author.

1.4 Thesis outline

The contribution of the rest of the chapters to the thesis is described under this topic. This thesis follows the manuscript format and each chapter are prepared as ‘standalone’ documents. Hence, chapter 2, chapter 3, and chapter 4 are allocated for the manuscripts submitted for review. In addition to the prediction methodology described in chapter 3, it utilizes the techniques developed in chapter 2 for the prognosis step. The complete software code used in studies presented in chapter 2 and 3 is presented as a separate manuscript in chapter 4 with links to the respective repositories.

As shown in *Figure 1-1*, the developments in chapter 2 and 4 are used to achieve the main goal of the thesis described in chapter 3. Chapter 2 mainly discusses the newly developed FDD system and how the two machine learning techniques are integrated to get a more robust system. It provides a detailed explanation of the methodology to detect and diagnose the root cause. Evaluation of the detectability and the ability to diagnose the actual root cause is essential for the next step, which is called prediction and prognosis.

Chapter 3 presents the manuscript developed based on fault prediction and prognosis. It provides a detailed explanation of the procedure to use HMM toolbox to predict a potential fault, based on the historical data of the system. This will be helpful to the researchers who will work with HMM tool box in variety of applications. Further, it uses the technique developed in chapter 2 to combine the two machine learning techniques to prognose the most probable root cause.

In chapter 4, the software code (CPL1.0) which is developed for the analysis done in chapter 2 and 3 is introduced as a manuscript. The strengths and potential applications are

described in detail. Further, all the important metadata are also provided with links for the repository of the software code.

As the final step, in chapter 6, the summary of the entire study is provided followed by a conclusion.

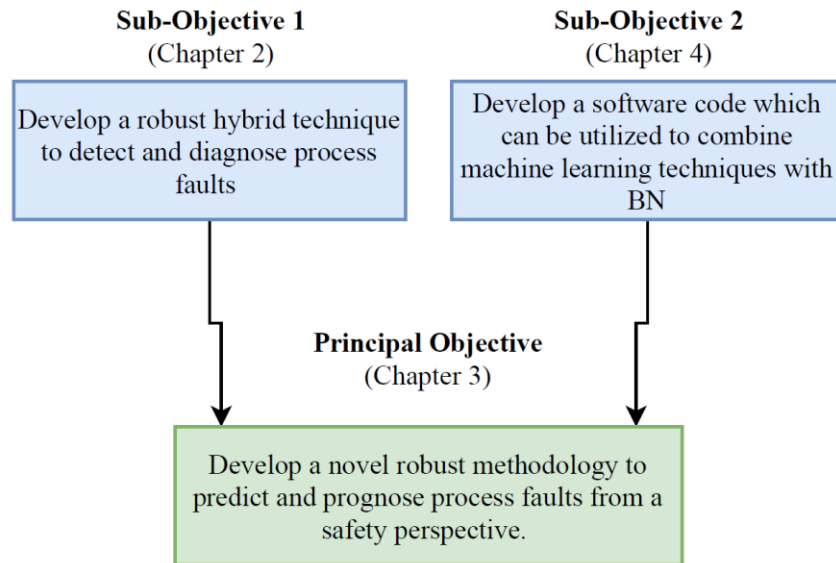


Figure 1-1: The contribution of each chapter to the formation of the thesis

CHAPTER 2 : DYNAMIC PROCESS FAULT DETECTION AND DIAGNOSIS BASED ON A COMBINED APPROACH OF HIDDEN MARKOV AND BAYESIAN NETWORK MODEL¹

Co-authorship Statement

Galagedarage Don Mihiran Pathmika is the principal author of this thesis. He has undertaken the research and prepared the first draft. Professor Faisal Khan, the co-author of the manuscript, shared the problem and conceptualized the methodology. In addition, Prof. Khan contributed by reviewing, and revising the manuscript. He also guided the author throughout the entire process of the methodology development, testing, validation and its application development. The software code and analysis of results were solely contributed by the principal author, and the results were validated for correctness by the co-author.

Abstract

The present study introduces a novel methodology for fault detection and diagnosis (FDD), based on a combined approach of data and process knowledge driven techniques. The Hidden Markov Model (HMM) is used to detect the abnormalities based on process history while the Bayesian Network (BN) diagnoses the root cause of the fault based. An HMM is trained with standard operating condition data while the structure of BN is developed based on process knowledge. The log-likelihoods (LL) of process history data string used to define the conditional probability tables of the BN. Abnormal behaviour of the system is identified through HMM. The time of detection of abnormality, respective

¹ This chapter is submitted as a manuscript to the journal of Chemical Engineering Science and currently under review

log-likelihood value, and the probabilities of being in the process condition, at the time of detection, are used as evidence to BN. The updated BN is then used to diagnose the root cause by considering the respective changes in the probabilities. Performance of the new technique is tested and validated with published data of Tennessee Eastman Process. Eight of the ten selected faults were successfully detected and diagnosed.

Keywords: *Process fault diagnosis; Hidden Markov model; Bayesian network, Process fault detection*

2.1 Introduction

Fault detection and diagnosis (FDD) is an essential component in process industries. As mentioned in (Mu and Venkatasubramanian, 2003), successful early detection and diagnosis of abnormalities in the petrochemical industry can save 20 billions of dollars per year, so-called the 'number one problem' that needs to be solved. This is considered as the core of Abnormal Event Management (AEM) which includes the detection, diagnosis, and correction of abnormal conditions of faults in a process. The main idea is to diagnose the fault before it reaches the un-correctable territory. Industrial practitioners and academic researchers have made a variety of approaches to solve this problem. It ranges from mathematical modelling to artificial intelligence (AI) and statistical approaches.

According to (Venkat *et al.*, 2003), no single method has all the desirable features for a diagnostic system. In their study, they have proposed a set of desirable characteristics that should be possessed by a diagnostic system. Then they have systematically evaluated a set of currently used diagnostic systems to reach the above conclusion. As they suggest, integrating the complementary features is one way to develop hybrid systems that could

overcome the limitations of individual solution strategies. (Mu and Venkatasubramanian, 2003).

Current work is planned with the objective to develop a novel robust methodology to detect and diagnose process fault from a safety perspective. This objective is achieved by integrating two techniques Hidden Markov Model (HMM) and Bayesian Network (BN). The HMM does the fault detection based on process history, while BN makes the fault diagnosis based on process knowledge. HMM also provides inputs to Conditional Probability Tables (CPT) of the BN. Further, these two techniques are amalgamated such that, the output of the HMM is fed to BN to do further analysis. Therefore, either method do work independently and hence no compromise on the performance of respective capabilities.

Section 2 of this paper present a comprehensive literature survey on fault detection and diagnosis using HMM and BN. Towards the end of the section, a detailed summary and knowledge have been identified. Section 3 provides details of the methodology which covers steps to detect the fault and diagnose the fault. Each step such as the training of HMM, generation of CPTs, generation of prior probabilities, and generation of likelihood evidence are described in detail. Section 4 is allocated for testing and validation of the proposed methodology. Results and Discussion are provided in section 5. Sample detection and diagnosis results are provided followed by a summary of results. Discussion includes the practical problems faced in detection and diagnosis and how they solved. A description of the coding procedure is also provided as a guide for the users. Finally, the conclusion is

presented in section 6 with descriptions of contributions made, the strengths of the proposed technique and future work.

2.2 Literature Survey

According to (Ding, 2014), the major and well-established technologies in FDD can be classified into the following categories, namely, Hardware redundancy-based fault diagnosis; Signal processing-based fault diagnosis; Statistical data-based fault diagnosis; Analytical model-based fault diagnosis; and Knowledge-based fault diagnosis.

As explained in (Ding, 2014), hardware redundancy-based fault diagnosis is a costly but highly reliable technique. The hardware redundancy is commonly used in mission and safety-critical systems such as digital fly-by-wire flight systems and nuclear reactors (Sawaragi, Soeda and Omatu, 1978). The cost goes up because it uses identical components to develop a redundant system. Comparison of the process component output with that of the redundant component is the concept in this approach. Although the concept is simple and straightforward, it is highly reliable as the fault can be isolated directly.

As further explained in (Ding, 2014), signal processing-based fault diagnosis is ideal for steady-state processes but has limited applications in dynamic processes. It can be performed in both the time domain and frequency domain. It looks for 'symptoms' by analyzing the changes in magnitudes and patterns of signals. Statistical data-based fault diagnosis techniques are also used for static process FDD. The main characteristic of this technique is, it requires process historical data to train a system and online real-time data to provide available evidence so that the trained system can provide useful outputs.

It is also mentioned in (Ding, 2014), analytical model-based fault diagnosis requires a mathematical model of the process being examined. The mathematical process model is always compared with the actual operation and generated a residual signal which is later analyzed to make decisions. The strength of this approach is it can even be used to analyze dynamic process systems. Knowledge-based fault diagnosis has a high potential being applied in FDD of complex processes. A qualitative model brings the prior knowledge of a process into the analysis. It consists of a knowledge base; a database; and inference engine; and an explanation component. BN is an example. As further explained in (Mu and Venkatasubramanian, 2003), the entire range can be categorized in various ways such as qualitative and quantitative; model driven, and entirely data-driven.

According to (Venkatasubramanian, Rengaswamy and Kavuri, 2003), a fundamental understanding of the underlying physics and chemistry involved with the process is essential for qualitative models. According to (Venkatasubramanian, Rengaswamy and Kavuri, 2003), there are two primary divisions namely topographic and symptomatic search techniques. A qualitative model of normal operation is required for topographic searches to perform malfunction analysis. On the other hand, symptomatic searches detect symptoms to direct the search to the fault location. Model-based approaches require a residual generator and a residual evaluator. The residual generator provides the residuals by comparing the plant output with the model output, and residual evaluator decides the faulty or average state of a process. Here the model can be a mathematical model. The data, or process history, based on techniques eliminate the requirement of a mathematical model. Instead, it generates correlations using the historical data. A significant amount of historical

data is transformed to construct the monitoring scheme for FDD. This transformation technique is known as the feature extraction. As mentioned in (Mu and Venkatasubramanian, 2003), data-based methods can be divided into two categories depending on the extraction process, namely, qualitative, and quantitative. Qualitative methods include the expert systems and qualitative trend analysis (QTA), while quantitative methods include the statistical tools PCA, ICA, ANN.

2.2.1 History of Data-based FDD

Dynamic risk assessment and fault detection technique is presented by (Zadakar, Imtiaz and Khan, 2013) using Principal Component Analysis (PCA). In this study, they have coupled PCA with quantitative operational risk assessment model to detect process abnormalities early. In addition to that, a novel algorithm to detect and diagnose some of the previously undetectable stochastic faults in the Tennessee Eastman (TE) process has been discussed by (Du and Du, 2018). This detection task is performed by the combined approach of Empirical Mode Decomposition (EEMD) with the PCA while the diagnosing is done using the Cumulative Sum (CUSUM). To enhance FDD performance, EEMD is combined with PCA as a pre-filtering tool in this work to extract fault signatures that can be further used to infer the occurrence of faults while CUSUM based statistics is especially suitable for detecting small changes in the process mean. Therefore, this combined approach has a better performance in comparison with pure PCA approach. (Du and Du, 2018). First, the measured variables are decomposed into different scales with the EEMD based PCA. Through this, the fault signatures can be extracted which can be used in FDD. CUSUM is used to minimize the detection delay. It is combined with T^2 and Q statistics.

In summary, they propose an enhanced approach based on multivariate and multiscale statistical analysis to improve the FDD; introduce an effective algorithm to detect and diagnose a specific set of faults which were undetectable with previous efforts; and successfully use empirical models to infer faults in chemical processes with dynamic changes between normal and faulty operating conditions.

A combined approach of Kernel Principle component analysis (KPA) and BN is presented by (Gharahbagheri, Imtiaz and Khan, 2017), which utilizes the diagnostic information given by KPCA, and the process knowledge acquired through BN. This technique was validated by applying in a benchmark process, and root cause diagnosis of abnormal conditions was successfully achieved.

A self-organizing map (SOM) based methodology is proposed by (Yu *et al.*, 2014) for FDD of processes with nonlinear and non-Gaussian features. The classification of state of the process is done concerning a SOM, trained with normal Operating Condition (NOC) data. The classification comes with a dynamic loading factor. The divergence of the dynamic loading factor is used to develop the contribution plots which is later used for fault diagnosis. Also (Yu *et al.*, 2014) introduces a new approach based on the Self-Organizing Map is proposed to detect and assess the risk of fault. The risk of fault is characterized using probabilistic analysis. The sensitivity of identifying the root cause of the fault is found to be comparatively high in the proposed method.

Further, (Yu *et al.*, 2014) have proposed a probabilistic multivariate fault diagnosis technique for industrial processes. The Gaussian copula is used to develop a dependence structure of the process variables and captures the non-linear relationships between process

variables. The online data string probabilities are calculated and compared with the defined limits and classified to faulty and non-faulty accordingly. Fault diagnosis is made based on reference dependence structures of the process variables which are determined from NOC data. These reference structures are then compared with those obtained from the faulty data samples and are a basis for fault diagnosis.

On the other hand, a Nonlinear Gaussian Belief Network (NLGBN) based fault diagnosis technique is proposed by (Yu *et al.*, 2014) for industrial processes. In this method, the features are extracted from process data using an NLGBN which is trained using NOC Data and a variational Expectation and Maximization algorithm. For fault diagnosis purposes, a multivariate contribution plot is also generated. The performance of this technique is found to be higher than that of conventional techniques such as KPCA, KICA, SPA, and Moving Window KPCA.

Also, (Onel *et al.*, 2018) have made a successful attempt to retrieve process measurements for FDD, utilizing high dimensional process data with nonlinear Support Vector Machine based feature selection algorithm. Further, as proposed by (Zadakbar, Imtiaz and Khan, 2012), PCA combined with quantitative operational risk assessment model can distinguish between operational deviations and abnormal conditions. The proposed method demonstrated better performance in early warning in comparison with univariate methods. An application of Logical Analysis of Data (LAD) to diagnose faults in industrial chemical processes is done by (Ragab *et al.*, 2018). In this approach, LAD discovers hidden knowledge in training datasets in the form of interpretable patterns that characterize the physical phenomena in process operation under normal or faulty conditions. The

discovered patterns are then combined to build a decision model that is used to interpret and to diagnose faults during the process operation. The proposed method performs better, in terms of accuracy, in comparison with, the Artificial Neural Network (ANN), Decision Trees (DT), Random Forests (RF), k Nearest Neighbors (kNN), Quadratic Discriminant Analysis (QDA) and Support Vector Machine (SVM) for the scenarios in consideration. A novel copula subspace division strategy is proposed by (Ren *et al.*, 2017) for FDD. Margin distribution subspace (MDS) modelled by joint margin distribution, and dependence structure subspace (DSS) modelled by copula are used to analyze high dimensional industrial data. The proposed methodology is found to be better in performance in comparison with PCA, ICA, and KPCA.

2.2.2 *Hidden Markov Model (HMM)*

According to (Rabiner, 1989), there are two types of signal or data models namely, Deterministic Models, and Statistical Models. In Deterministic Models, all that required is to determine values of the parameters of the data model such as amplitude, frequency, the phase of a sine wave, amplitudes and rates of exponentials. On the other hand, Markov and Hidden Markov Process, Gaussian Process, and Poisson Process can be introduced as Statistical Models.

As further explained in (Rabiner, 1989), the HMM is favourite in a wide variety of applications due to two strong reasons. The models are very rich in mathematical structure and hence can form the theoretical basis for use in a wide range of applications; and the

models, when appropriately applied, work very well in practice for several critical applications such as speech recognition.

To make the HMM model to be useful in the real-world applications, there are three main problems to be solved (Rabiner and Juang, 1986). This problem-solving approach for characterizing the theoretical aspects of hidden Markov modelling is introduced and successfully used by Jack Ferguson of IDA (Institute for Defense Analysis). (L. R. R. Rabiner, 1989)

The first problem is the evaluation of the probability (or likelihood) of a sequence of observations given a specific HMM. In other words, Given the observation sequence $O = O_1 O_2 \dots O_n$ alternatively, and a model $\lambda = (A, B, \pi)$, the method to efficiently compute the probability of the observation sequence. i.e. $P(O|\lambda)$. In simple terms, it is a problem of determining the most likely state path, given a sequence of emissions.

The second problem is the determination of the best sequence of model states. In other words, given the observation sequence $O = O_1 O_2 \dots O_n$ alternatively, and the model λ , the method of choosing a corresponding state sequence $Q = Q_1 Q_2 \dots Q_r$, which is optimal. In other words, it is the best “explains” the observations.

In practical applications, we cannot find a hundred percent correct state sequence, but an optimum sequence. The disadvantage is, there can be many reasonable optimality criteria can be applied in one case, and the result strongly depends on the criteria we selected.

The third problem is to adjusting model parameters so that it best accounts for the observed signal. In other words, it is adjusting the model parameters $\lambda = (A, B, \pi)$ such that maximize $P(O|\lambda)$. Once the optimum state sequence is determined, the model parameters

can be fine-tuned to make it the best explanation for a given observation sequence. The process of adjusting the model parameters using an observation sequence is called ‘Training’ of the HMM. This is the most crucial step in creating the best model for real phenomena.

2.2.3 Mathematical Formulation

As described in (de Almeida & Park, 2008), HMMs are a particular kind of Bayesian Networks (BN). The factorization of 1st order HMMs joint probability distribution is represented by Equation [1] where, $q_{1..T} = \{q_1, q_2, \dots, q_T\}$ is a sequence of states, $o_{1..T} = \{o_1, o_2, \dots, o_t\}$ is a sequence of observations or outputs, and t is an integer valued index.

$$P(q_{1..T}, o_{1..T}) = P(q_1)P(o_1|q_1) \prod_{t=2}^T P(q_t|q_{t-1})P(o_t|q_t) \quad [1]$$

As further explained in (de Almeida & Park, 2008), once the state-transitions rule follows the Markov property, the HMM concept also can be considered as an extension of Markov chains. Here Markov property means q_t depends only on q_{t-1} . The hidden term in HMMs is exactly due to its introduction since the underlying sequence of states, i.e. the Markov chain, is not directly observable. sequence of states, i.e. the Markov chain, is not directly observable. The factorization of the joint probability distribution in Equation [1] based BN representation is depicted in Figure 2 (de Almeida & Park, 2008).

The parameters required to define discrete HMMs are presented in *Table 2- 1*. Here M_D is the number of distinct observation symbols in the emission probability distributions, and N is the size of the discrete state space. λ is the compact notation for the above parameters. Hence, $\lambda = (\pi, A, B)$ (de Almeida & Park, 2008).

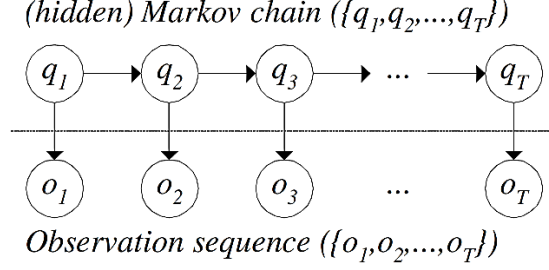


Figure 2: First order HMM in the form of BN representation (de Almeida & Park, 2008)

Table 2-1: Discrete HMM elements (de Almeida & Park, 2008)

Parameters	Description
$A = \{a_{ij}\}$	State transition probability distribution $a_{ij} = P(q_{t+1} = j q_t = i), 1 \leq i, j \leq N$
$B = \{b_j(k)\}$	$b_j(k) = P(o_1 = v_k q_t = j), 1 \leq k \leq M_D, 1 \leq j \leq N$
$\pi = \{\pi_i\}$	Initial state distribution $\pi_i = P(q_1 = i), 1 \leq j \leq N$

In the continuous case, probability density functions replace the B matrix. Usually they are represented as a finite mixture of Gaussian distributions which is described in Equation [2]. Here, o_t is the observation vector, M_c is the number of mixture components, μ_{jk} is the mean vector, and Σ_{jk} is the covariance matrix, for the k th mixture component in the state j .

$$b_j(o_t) = \prod_{k=1}^{M_c} c_{jk} N(o_t, \mu_{jk}, \Sigma_{jk}), \quad 1 \leq j \leq N \quad [2]$$

$$\sum_{k=1}^{M_c} c_{jk} = 1, \quad 1 \leq j \leq N \quad [3]$$

$$c_{jk} \geq 0, \quad 1 \leq j \leq N, 1 \leq k \leq M_c \quad [4]$$

2.2.4 HMM-based FDD

As mentioned in (Li *et al.*, 2005), HMM has a variety of strengths. HMM is very suitable for modelling the dynamic time series, and has a robust capability of pattern classification, especially for a signal with abundant information, non-stationarity, poor repeatability and reproducibility. At the same time, HMM can process the long random sequences in theory. They have used HMM as a classifier of features derived by Fast Fourier Transform (FFT), wavelet transform, spectrum. The proposed approach mainly targets speed-up and speed-down process in rotating machinery and have shown that it is useful and efficient to be used in the same area. Also, a fascinating study has been done by (Boyras, Acar and Kerr, 2007) to detect abnormal driving patterns of a person by observing the movement pattern of a vehicle.

Conventional FDD techniques find it is challenging to diagnose faults in processes with multiple operational modes and transitions. A novel FDD method is proposed by (Wang *et al.*, 2016) which works based on HMM. Here they use two different HMM models for steady, and transition processes respectively. The Bayesian information criterion (BIC) does the model evaluation. After an appropriate model is acquired, an index named negative log likelihood probability is employed for transition process fault detection.

A modified HMM (MHMM) is successfully used by (Lee *et al.*, 2010) to diagnose the degradation processes of multiple failure modes. To rapid detection of an abnormality, MHMM is coupled with statistical process control. The proposed methodology has been successfully applied in detecting tool wear states of known states and in addition to that the unknown tool wear states also can be identified in the early stages. Further, (de Almeida

and Park, 2008) have proposed a method to investigate both unexpected and incipient faulty events. To the former, detection and diagnosis tasks were immediately satisfied; and to the latter, they were carried out in a progressive and correct course.

Condition Based Maintenance (CBM) is a useful maintenance method as it minimizes downtime of a process. A method was proposed by (Choi and Yoo, 2014), as a pattern recognition tool, to detect, locate, and quantify structural flaws such as cracks. Again, in this study, FFT is used as a feature extraction tool.

PCA is a well-known feature extraction tool while HMM is a good classifier. Both these strengths are combined for FDD by (Choi and Yoo, 2014). The moving window for tracking dynamic data is used and have come up with useful results. A similar study is presented by (Huang and Zhang, 2009), for FDD of Diesel Engines. Here they have used PCA and Discrete HMM (DHMM) for classification. Nevertheless, PCA is not a successful feature extraction tool in extracting nonlinear relationships among process variables. Therefore, (Wang *et al.*, 2015) have proposed a switched feature extraction procedure using PCA and KPCA based on nonlinearity measure.

Independent Component Analysis (ICA) is also another powerful tool in FDD. Several methods are proposed in the literature using a combined approach of ICA and HMM. Nevertheless, (Li *et al.*, 2006) has proposed a methodology called ICA-FHMM (Independent Component Analysis-Factorial HMM) which has shown superior performance over ICA-HMM. In the proposed method, ICA is used for the redundancy reduction and feature extraction of the multi-channel detection, and FHMM as a classifier to recognize the faults of the speed-up and speed-down process in rotating machinery.

Wong & Lee (2009) proposed a technique for abnormality detection in financial applications based on HMM. They have presented a detailed performance analysis of the proposed abnormality detection algorithm, along with a comparison with the maximum likelihood-based data mining method. This method detects abnormalities in financial applications while giving minimum false alarms. Further, they have presented algorithms to as a solution for dynamic multiple fault diagnosis (DMFD) problems based on HMM. They have solved each of the DMFD problems by combining Lagrangian relaxation and the Viterbi decoding algorithm iteratively. Also (Cao *et al.*, 2015) have introduced a method to detect abnormal deviations of market prices (i.e. price manipulations). In this approach, Wavelet Transformations, and gradients are taken as the feature extraction methods to support Adaptive HMM with Anomaly States Model (AHMMAS) to detect price manipulations.

Forward-backward (FB) procedure and the Baum-Welch (BW) algorithm are used for parameter estimation in HMMs which make the computation considerably complicated. To address this issue (Li, Fang and Xia, 2014) introduced an increasing mapping based HMM, which is called IMHMM, which needs a lower storage requirement, and training time than that of HMM. Moreover, the higher performance was observed in comparison with PCA.

For FDD, PCA and ICA are not perfect choices always for chemical processes with multiple operating conditions and system uncertainty. An HMM-based ICA approach is proposed by (Li, Fang and Xia, 2014) for fault detection. Trained HMM along with the localized ICA models to identify various states of operation. Classification of operation

modes is done by HMM-based state estimation. Further, HMM is built from measurement data to estimate dynamic mode sequence. After that, the localized ICA models are developed to characterize various operating modes adaptively. HMM-based state estimation is then used to classify the monitored samples into the corresponding modes, and the HMM-based I^2 and SPE statistics are established for fault detection.

HMM is widely used in stock market predictions and abnormality detection. A combination of HMM with Fuzzy models is presented by (Hassan, 2009) to identify similar data patterns from history to predict future market changes. The next day market behaviour has predicted using a weighted average and Fuzzy Logic to forecast the value. Log-likelihood for a given data string can be introduced as a measurement for the compatibility to the trained HMM with a given set of data. This log-likelihood value is used to generate a fuzzy rule so that the value of the stock market can be predicted. It has shown superior performance in comparison with Auto-Regressive Integrated Moving Average (ARIMA), and Artificial Neural Network (ANN).

HMM, and its combinations are widely used in FDD in mechanical systems such as bearings, gear systems, and other rotating machinery. A Mixture of Gaussians HMM for failure diagnostic and prognostic is presented by (Tobon-Mejia *et al.*, 2010), which is tested for benchmark data related to bearings. In this study, the off-line component is mainly about feature extraction from the sensor outputs and train the models. In an online component, the learned models are used to diagnose failures by estimating the asset's current health state, its remaining useful life and the associated confidence degree. Same models can be used to prognosis the faults. In (Wang *et al.*, 2009), Wavelet Transform

(WT), Wavelet Lifting Scheme (WLS) and Empirical Mode Decomposition (EMD), are used for feature extraction. Thereafter, Singular Value Decomposition (SVD) is utilized to extract an intrinsic characteristic of the signal from the obtained matrix. These singular value vectors are regarded as inputs to HMM for system FDD. The classification rate found to be excellent in bearing FDD. Also, (Sadhu, Prakash and Narasimhan, 2017) proposes a method which involves preprocessing of data to improve the sensitivity of HMM classification. In their approach, wavelet transformation is used to extract features from denoised data. The decision tree is used to extract the most relevant data and a Gaussian mixing model-based HMM is then employed for fault detection. The proposed technique has a better performance over the traditional HMM in multiple fault states. On the other hand, (Yuwono *et al.*, 2016) have also proposed a methodology based on HMM and Swarm Rapid Centroid Estimation (SRCE) to detect bearing faults automatically. Here also defect frequency signatures are extracted with Wavelet Kurtogram and Cepstral Liftering. Neighborhood Component Analysis (NCA) and Coupled HMM techniques are combined with the methodology proposed by (Zhou *et al.*, 2016). The experiment results show that the proposed NCA-CHMM can remove redundant information, fuse data from different channels and improve the diagnosis results.

Also, (Soualhi *et al.*, 2012) has proposed a method to diagnose faults in induction motors using HMM. An HMM is trained for each fault type, based on the current and voltage data that it draws during operation. While in operation, the features of the voltage and current readings are matched with the trained models and the classification is done based on it. The experiment results prove the efficiency of the proposed method in comparison with

techniques such as neural-networks based approaches. The fault identification of induction motors by HMM has been further improved by (Yusuf *et al.*, 2013), with the introduction of a secondary classification tool called Naïve Bayes classifier.

According to (Yusuf *et al.*, 2013), fault diagnosis approaches of systems can be divided into three crucial areas namely, physical based model, AI based model and data-driven based model. As further explains, the first type model requires specific mechanistic knowledge and theory relevant to the monitored system structure which is hard to realize; and the second type model needs massive amounts of condition monitoring data which are also not always available; while data-driven model investigate proper statistical model to describe system state which is used widely in fault diagnosis domain. Among many data-driven techniques, (Jia, Sun and Teng, 2012) compares the strengths of Particle Filtering method and HMM method. As they conclude, particle filtering method has better detection performance, while HMM has better computation efficiency in the area of gearbox fault detection.

The performance comparison between HMM and GMM in bearing fault classification is presented by (Nelwamondo, Marwala and Mahola, 2006). The time-domain vibration signals of a rotating machine with standard and defective bearings are processed for feature extraction. Both linear and non-linear features are extracted using two feature extraction techniques. The extracted features are then used to classify faults using Gaussian Mixture Models (GMM) and HMM. The results show that HMM outperforms GMM in the application area of bearing fault classification.

As described by (de Almeida & Park, 2008) The reliability of an event (λ) classification increases with the use of observed data (o) in comparison with the sole use of prior probability ($P(\lambda)$). The conditional probability can be determined using Bayes Rule. Further, $P(\lambda)$ is the likelihood of occurrence of λ with respect to o . As the probability distribution of the data ($P(o)$) is independent of λ , Equation [5] can be presented as Equation [6]. In a fault detection operation, it is considered a characteristic normal operation (i.e. $P(\lambda) = 1$). The detection is made based on the observed data. In other words, it is the likelihood function ($P(o|\lambda)$), which is exactly the output of HMMs.

$$P(\lambda|o) = P(o|\lambda)P(\lambda)/P(o) \quad [5]$$

$$P(\lambda|o) \propto P(o|\lambda)P(\lambda) \quad [6]$$

When the HMMs are used in diagnosis applications, the winner HMM (λ^*) is the one with a maximum value of $P(o|\lambda)P(\lambda)$ for the models (λ) where $\lambda^* = \max [P(o|\lambda)P(\lambda)]$. Therefore, it is clear that the main target of HMM is to model sequential data. As shown in Figure 3, the input is a temporal sequence of T vectors ($O = \{o_1, o_2, \dots, o_T\}$), and the output is a likelihood value ($-\log[P(O|\lambda)]$) which measures the compatibility of the model (i. e. λ) in generating the observed data (O). Therefore, this technique can be identified as a sequential pattern recognition tool. (de Almeida & Park, 2008)

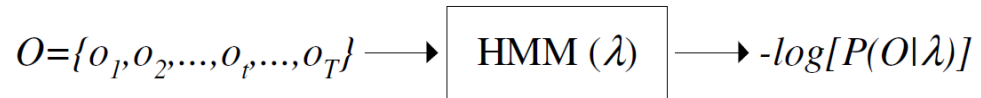


Figure 3: Input-output relationship of HMM (de Almeida & Park, 2008)

A summary of contributions discussed under this topic in journals is presented in *Table 2-2* and it is clear that still a considerable amount of hybrid options to be explored such as HMM-BN hybrid FDD.

2.2.5 HMM Tool Box for MATLAB

Murphy (2005) provide open source toolbox support inference and learning for HMMs with discrete outputs, Gaussian outputs, or mixtures of Gaussians output. The Gaussians can be full, diagonal, or spherical. It also supports discrete inputs. On the other hand, the inbuilt HMM toolbox in Matlab supports discrete input and outputs. It requires state probabilities and transition probabilities to be defined or generated by the user hence have limited capabilities.

Table 2-2: Summary of HMM-based fault diagnosis

Author(s)	Journal	Method/Model
Li, Z., Wu, Z., He, Y., & Fulei, C. (2005)	MSSP	HMM
Li, Z., He, Y., Chu, F., Han, J., & Hao, W. (2006)	JSV	ICA and Factorial HMM
Nelwamondo, F. V., Marwala, T., & Mahola, U. (2006)	ICIC	HMM, GMM, mel-frequency cepstral coefficients and fractals
de Almeida, G. M., & Park, S. W. (2008)	IFAC	HMM
Hassan, M. R. (2009)	Neurocomputing	HMM and fuzzy model
Wong, W. C., & Lee, J. H. (2009)	IFAC	HMM
Lee, S., Li, L., & Ni, J. (2010)	JMSE	Modified HMM
Chen, Z., & Yang, Y. (2012)	JVA	Multi-Sensor Mixture HMM
Rashid, M. M., & Yu, J. (2012)	IECR	HMM Based Adaptive ICA
Choi, C. K., & Yoo, H. H. (2014)	IJPEM	HMM
Li, Z., Fang, H., & Xia, L. (2014)	ESA	Increasing mapping based HMM
Cao, Y., Li, Y., Coleman, S. A., Belatreche, A., McGinnity, T. M. (2015)	IEEE TNS	Adaptive HMM
Wang, L., Yang, C., Sun, Y., Pan, Y., & An, R. (2015)	JP	HMM
Wang, F., Tan, S., Yang, Y., & Shi, H. (2016)	IECR	HMM
Yuwono, M., Qin, Y., Zhou, J., Guo, Y., Celler, B. G., & Su, S. W. (2016)	EAAI	Particle swarm clustering and HMM
Zhou, H., Chen, J., Dong, G., Wang, H., & Yuan, H. (2016)	MSSP	NCA and coupled HMM
Sadhu, A., Prakash, G., & Narasimhan, S. (2017)	JVC	HMM

2.2.6 BN based FDD

An offline fault diagnosis method for industrial gas turbines in a steady-state is presented by (Lee *et al.*, 2010). They have shown that the accuracy can be increased by using multiple Bayesian models which are trained to identify specific faults. Also, that method can handle more than one faults occurring in more than one component. Further, it can identify random faults and systemic faults such as sensor bias.

A method of BN fault diagnosis in the satellite power system is presented by (Xie, 2013). In establishing the CPT of BN, it uses an algorithm to adopt a statistical strategy for the rule library provided by many experts, extracts a causal relationship from the expert knowledge base. It is shown that the proposed technique is useful in fault diagnosis of satellite power systems.

Even based on the uncertain knowledge and incomplete information, BNs can perform better than back-propagation neural networks and probabilistic neural networks. The method is proposed by Cai, Huang, & Xie (2017) in the application area of gear train systems.

Object-oriented Bayesian networks (OOBNs) is applied in a scenario of the subsea production system by (Cai, Huang and Xie, 2017) and shown improved performance. This methodology can be used in real time fault detection and diagnosis. There are two phases namely off-line and On-line. During the off-line OOBN construction phase, historical sensor data and expert knowledge are collected and processed to determine the faults and symptoms, and OOBN-based fault diagnosis models are developed subsequently. In the

on-line phase, operator experience and sensor real-time data are placed in the OOBNs to perform the fault diagnosis.

In (Chan and McNaught, 2008), the authors employ Bayesian networks (BNs) to model the domain knowledge that comprises the operations of the System Under Test, and the diagnostic skill of experienced engineers. This enhances the efficiency and reliability of the diagnostic process. This diagnostic tool is named 'Wisdom,' which is applied in the area of manufacturing tests of mobile telephone infrastructure.

An exciting review is done by (Cai, Huang and Xie, 2017) which presents the flowchart in *Figure 2-4*. In this review, they insist that future research should focus on hybrid approaches with BN and other fusion techniques. Further, (Cai, Huang and Xie, 2017) presents information on Fault Diagnosis for Process Systems; Energy Systems; Structural Systems; Manufacturing Systems; and Network Systems.

As further mentioned in (Cai, Huang and Xie, 2017), attempts for fault detection and diagnosis made in various process engineering applications can be listed as shown in *Table 2-3*. As ongoing and up-coming research directions, they introduce, Integrated Big Data and BN Fault Diagnosis Methodology; BN-Based Non-permanent Fault Diagnosis; Fast Inference Algorithms of BNs for Online Fault Diagnosis; BNs for Closed-Loop Control System Fault Diagnosis; Fault Identification Rules; and Hybrid Fault Diagnosis Approaches.

2.3 Main observations from the Literature Survey

It is highly unlikely to have a pure single method to provide robust detection and diagnosis performance. Therefore, experts recommend taking the path of hybrid approaches to

improve the accuracy of detection. Further, HMM is a powerful tool in the classification of Faults in a vast area of applications.

Moreover, HMM has a strong theoretical background. Also, some powerful open source toolboxes, with proper documentation, are available for advanced data analysis with HMM. On the other hand, BN has a proven ability to diagnose the root cause of faults in a wide variety of applications. It can bring the process knowledge, expert judgements and experience into the FDD process. Considering the above facts, a hybrid method comprising of HMM, and BN is proposed. This combined approach will strengthen the FDD process by incorporating the process history, and the process knowledge.

Table 2-3: FDD in process engineering applications

Author	Journal	Proposed Method and Application
Qi & Huang (2011)	Automatica	Bayesian-method-based fault node diagnosis approach for control Loop: Distillation column process
Jin, Liu, Lai, Li, & He (2017)	IJAMT	Ceramic-shell deformation root-cause analysis and fault diagnosis
Liu, Zhang, & Shi (2014)	IEEE TSE	BN-based process monitoring and fault diagnosis approach to study sensor allocation methods for process control application
Verron, Tiplica, & Kobi (2010)	EAAI	A novel BN based fault diagnosis approach for industrial process control.
Dos Santos, Ebecken, Hruschka, Elkamel, & Madhuranthakam (2014)	Risk Analysis	BNs as a classifier to detect and diagnose faults in process systems

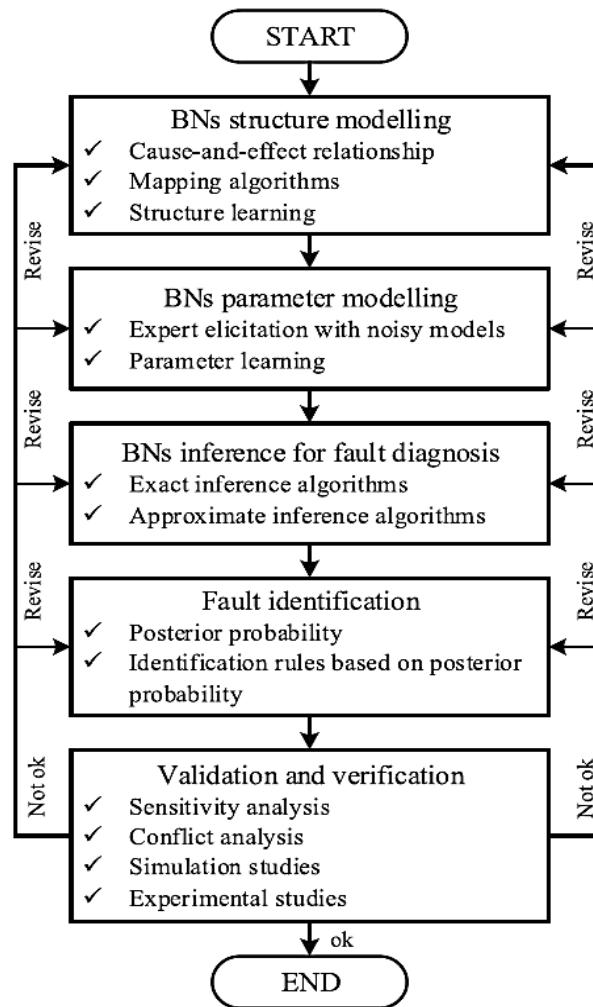


Figure 2-4: Flowchart of BN based Fault Diagnosis (Cai, Huang and Xie, 2017)

2.4 The Methodology

The outline of the FDD approach is discussed under this topic which is graphically represented in *Figure 2-2*. HMM is employed to extract information from the process history, which is trained with NOC data. In addition to the process history, the process knowledge is also used to make the detections and diagnosis more accurate. BN is employed to extract information from process knowledge. A systematic approach, using

Sign Directed Graphs (SDG), is used to develop the structure of BN. Moreover, the CPTs of BN are filled with the use of HMM outputs in NOCs. Prior probabilities of the BN are also determined using the output of HMM.

In the fault diagnosis step, the likelihood evidence is generated for incoming or testing data strings, using the same HMM, and then the BN is updated accordingly. After updating with likelihood evidence, the node in the BN with the maximum change in the probability of failure is identified as the cause if it is a root node. If not, the highest percentage in the preceding successive node is selected as the root cause.

Two different data types were used namely, training data of NOCs (TRD_1), and testing data of operating conditions with faults (TED_1). Initially, an HMM (HMM_1) was trained using the training data set with NOCs. The trained HMM_1 was then used to generate a history of LL values for each data string of TRD_1. This data history with LLs is used to calculate the conditional probabilities which are later used to prepare the CPTs of the BN. In other words, this BN is trained with the NOCs outputs of HMM_1. Further, the prior probabilities were calculated based on the history of LL values using HMM_1 which is later used in parent nodes of BN.

2.4.1 Illustration of the methodology using tank model

The simple tank model illustrated in Figure 2.6 will be employed to illustrate the methodology developed for FDD of process system faults. The primary measurable variables are F_1 and F_2; where F_1 has a positive effect on the level (L), and F_2 has an adverse effect on L.

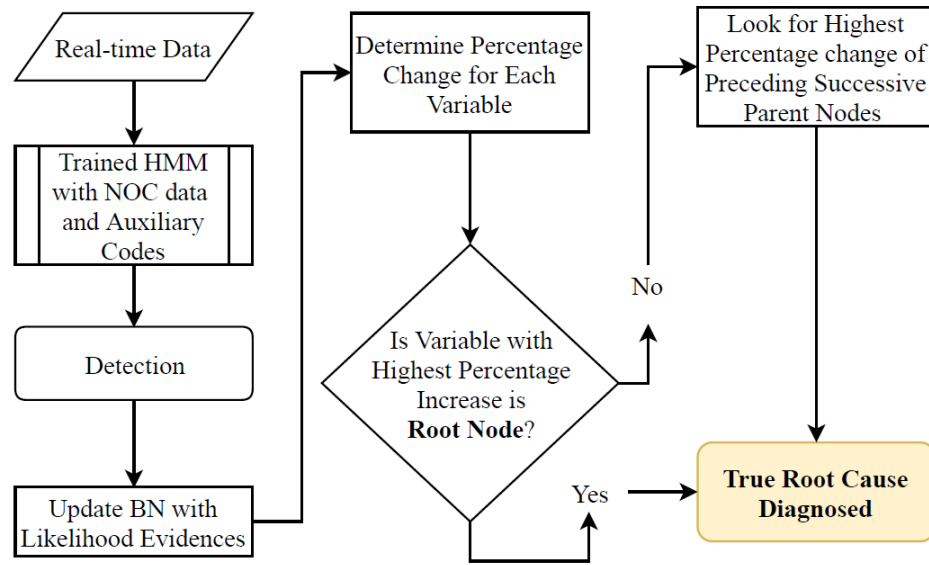


Figure 2-5: The overall procedure of FDD

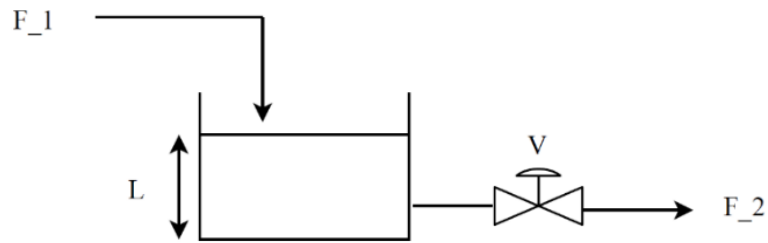


Figure 2-6: Tank model

2.4.2 Training of HMM₁

The first half of TRD₁ (NOC) is used to train the HMM₁. Many hidden states (Q), number of mixtures of Gaussians (M) were set such that they give the maximum detection accuracy. K-fold cross-validation was used to determine the optimum number of hidden states. Here, the total data set is separated in to (say N) sets. Then any $N-1$ sets were selected as training data, and the remaining one can be used for validation. This value of N

can be varied over a possible range, and the mean and standard deviation can be used in making the decision.

2.4.3 Development of the Bayesian Network

Under this section, the development of BN is illustrated for the defined scenario.

Figure 2-7 illustrates the procedure to develop the quantitative BN. Initially, the structure of the BN is developed based on the process knowledge. Then it is trained using the output of HMM_1.

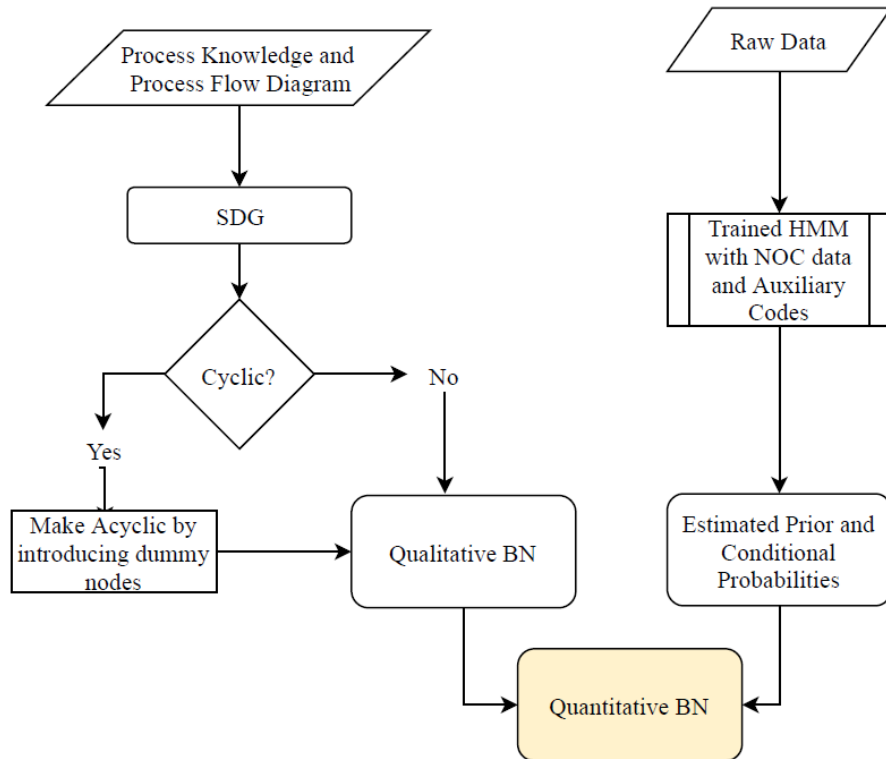


Figure 2-7: Procedure to develop the Quantitative BN

2.4.4 *Development of Qualitative BN*

Based on (Mallick, 2013), the steps for developing the SDG are presented under this topic. Firstly, the random variables involved with the process are identified. (i.e. F_1 , F_2 , V , and L where, inflow, outflow, valve resistance, and tank level respectively). Then the causalities are identified and illustrated using the arrow and respective sign. For example, consider level (L), and inflow (F_1). F_1 can cause an increase of Level (L). In other words, when F_1 is increasing, L will receive a positive effect. It can be graphically presented as shown in *Figure 2-8*. The inverse is also true. As further shown in *Figure 2-8*, the SDG is cyclic between F_2 and L , which makes it impossible to map into a BN which is acyclic. Therefore, the modification shown is introduced. To keep the feedback into L from F_2 , a recycled F_2 (i.e. F_{2_R}) is introduced as a new variable. This is a duplicate dummy node introduced in order to keep the acyclic nature of the BN while keeping the feedback from F_2 .

2.4.5 *Mapping the SDG to BN*

According to (Mallick, 2013), the following algorithm can be used to map SDG to BN. As further explained, after developing SDG, it is mapped to the Bayesian Belief Network (BBN) based on both graphical and numerical translation. The structure of BBN is obtained from the graphical translation. The nodes are connected in the same way as they are connected in the SDG. The root nodes, intermediate nodes and effect nodes are mapped into the BBN as parent nodes, intermediate nodes and child nodes. The mapped BN is shown in *Figure 2-10*.

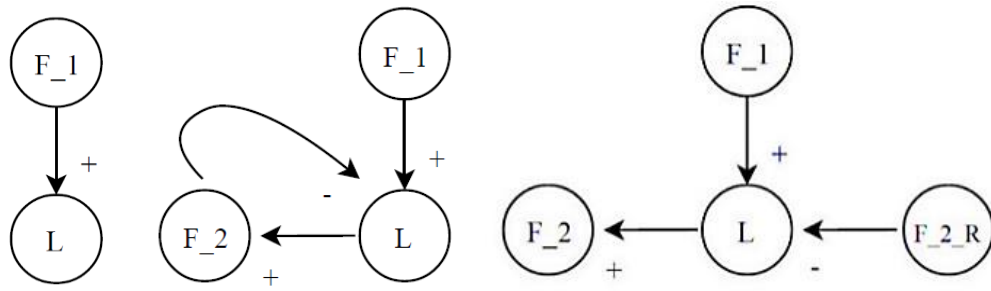


Figure 2-8: Development of SDG

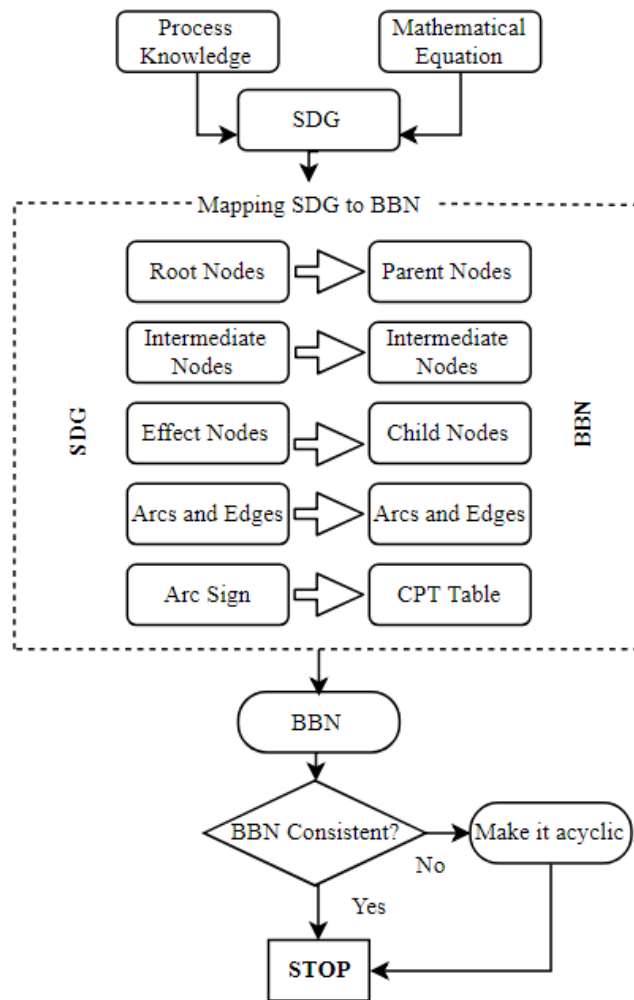


Figure 2-8: Mapping of SDG to BN (Mallick, 2013)

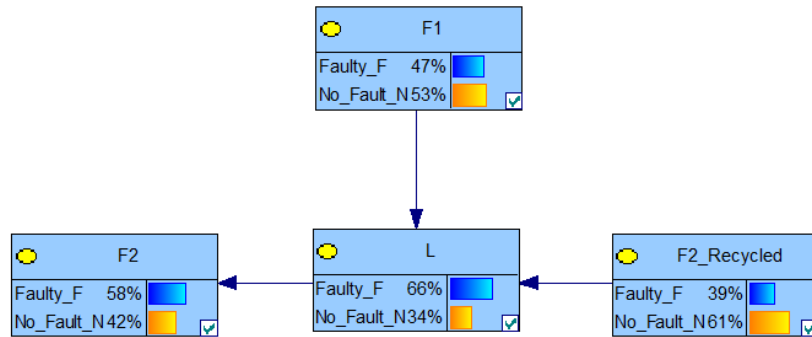


Figure 2-10: The BN representing the Tank Model

2.4.6 Training the BN

This is an essential step in the entire study. First, as shown in

Figure 2-10, all values in NOC data are classified into the following 3 zones using Matlab code. Safe Zone was given the designation 'N' with the meaning 'No-Fault.' Because it is less likely to fault a particular parameter if it is inside the Safe Zone. Also, the Danger Zone was given the designation 'F' ('Faulty') as it is more likely to fault a particular parameter if it is inside the Danger Zone. Further, the constants aNL and aNH are the lower limits and Higher limit of the safe zone respectively.

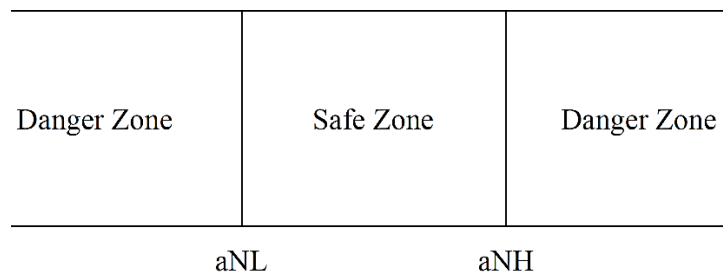


Figure 2-10: Different Zones in NOC Data

The constants aNL and aNH were calculated using the equations [7] and [8].

$$a1L = \text{mean}(A(:,1)) + (\text{min}(A(:,1)) - \text{mean}(A(:,1)))/r \quad [7]$$

$$a1H = \text{mean}(A(:,1)) + (\text{max}(A(:,1)) - \text{mean}(A(:,1)))/r \quad [8]$$

Here, $A(:,1)$ stands for the 1st column of the matrix A . Matrix A represents the concatenated matrix of past NOC data and the respective LL values generated through HMM. The value of r was selected such that it gives a proper distribution of data in the safe zone and danger zone. Based on aNL , and aNH the entire data set is classified according to all possible sequences. A sample output of the HMM can be illustrated as follows.

Sequence	Log-likelihood(LL)
$F N F F$	2.5×10^{-5}

This means, in the NOC data set, this is a data string where 1st, 3rd, and 4th variables are in the Danger Zone while the 2nd variable is in the Safe Zone. The whole idea of this classification into zones is to generate a set of events and their likelihood values through HMM. This outcome is used to calculate the CPTs of each node in BN. As it can be seen in the sample output, the LL is generally a minimal value. This LL is the parameter which carries the information of probability of a given sequence. As it is purely not a probability value, following mathematical procedure can be followed to normalize it and generate probability values to establish the CPTs.

If a relative precision of ϵ is required (say, $\epsilon = 10^{-d}$ for d digits of precision) and if have n likelihoods are available, any result less than the logarithm of $\frac{\epsilon}{n}$ can be eliminated.

Then it can proceed as usual to exponentiate the resulting values and divide each one by

the sum of all the exponentials. In mathematical terms, let the log-likelihood values be $\lambda_1, \lambda_2, \dots, \lambda_i, \dots, \lambda_n$ with $\lambda_n = \max(\lambda_i)$. For the logarithm to the base $b > 1$.

Define;

$$\alpha_i = \begin{cases} b^{\lambda_i - \lambda_n}, & \lambda_i - \lambda_n \geq \log(\epsilon) - \log(n) \\ 0, & \text{Otherwise} \end{cases} \quad [9]$$

The normalized likelihoods equal to; α_i/A

$$A = \sum_{j=1}^n \alpha_j \quad [10]$$

Where, $i, j = 1, 2, 3, \dots, n$

This technique is applicable because by replacing all of the otherwise under flowing α_i by zero makes a total error of at most $\frac{(n-1)\epsilon}{n} < \epsilon$. As because $\alpha_n = b^{\lambda_i - \lambda_n} = b^0 = 1$ and all α_i are non-negative, the denominator $A = \sum_j \alpha_j \geq 1$. Hence, the total relative error due to the zero-replacement rule is minimal then $((n-1)\epsilon/n)/A < \epsilon$ as desired.

Based on the above normalizing method, the entire detection history can be classified into all possible combinations of events and their respective probabilities can be determined. Prior probabilities were also determined based on the detection history generated through HMM_1. Matlab codes are developed to do all the calculations. The code can be extended to complicated BNs by following the pattern.

2.4.7 Introduction of Likelihood Evidence to the trained BN

There can be an inherent delay in fault detection by HMM due to the physical and chemical behaviour of the process being observed. Therefore, in calculating the likelihood evidence, data strings in the recent past are also considered. This assumes that HMM detects the fault

after t_0 number of seconds. The value of t_0 depends on the complexity of the fault. If the LL value clearly shows a deviation soon after introducing the fault, t_0 can be a minimal value and vice versa. An output similar to *Figure 2-11* can be taken through HMM. The Matlab function ‘findchangepts’ can be employed to detect the point where the abnormal conditions start appearing.

Once the fault is detected, the respective time (t_f) and data string can be determined. Based on the values of aNL , aNH , and data strings from $t_f - t_0$ to t_f , the probability of giving $F F F F$ sequence at $t = t_f$ is calculated with Matlab code. The respective probabilities of respective nodes are taken as the likelihood evidence for the BN. As mentioned in *Figure 2-5*, The fault is diagnosed.

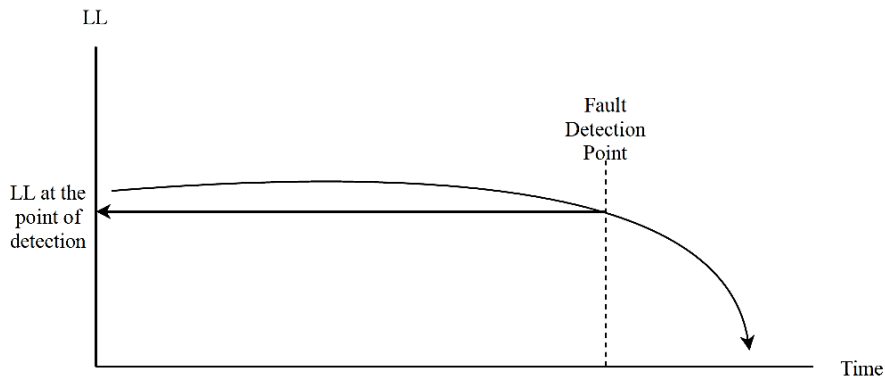


Figure 2-11: Output of HMM

2.5 Application of the methodology in Tennessee Eastman (TE) Process

2.5.1 Tennessee Eastman Process

The TE chemical process produces a broad range of advanced materials, chemicals and fibres for everyday purposes. It has five significant units namely; a reactor, a product condenser, a vapour-liquid separator, a recycle compressor and a product stripper. Three

gaseous reactants are fed to the reactor, where a catalyzed chemical reaction forms the liquid products. The product stream enters the condenser as vapour and gets condensed. Then product stream passes through the vapour-liquid separator, where the condensed and non-condensed products are separated. A centrifugal compressor recycles the non-condensed product back to the reactor, and the condensed product moves into the stripper to be stripped. The final product stream exits from the base of the stripper and is pumped to the downstream for further refinement (Downs and Vogel, 1993).

The PFD of the TE chemical process is shown in *Figure 2-10*. The TE chemical process consists of 41 measured variables and 12 manipulated variables. Among the measured variables, 22 variables are continuous process variables, and 19 variables are related to composition measurements. These 22 continuous process variables have been considered in this work, and their description is shown in the *Table2- 4*. There are 15 known and five unknown types of faults in the TE chemical process (Downs and Vogel, 1993); (Yu, Khan and Garaniya, 2015). Among them, ten widely studied fault scenarios are tested. This study is focused on these ten faults because of the availability of data and to compare the effectiveness of the methodology against previous studies. The tested fault IDs and their true root causes for each fault type are summarized in *Table2- 5*.

2.5.2 *Fault detection by HMM*

Similar to the illustration given in 0, the HMM_1 is trained using the NOC data extracted from all ten faults in TRD_1. The idea is to detect the change in LL value once the incoming data string becomes abnormal.

It is normal to have some fluctuations of LL value due to the physical and chemical behaviour of the system. Due to this reason, there are some faults which cannot be identified as soon as introduced to the system. Most of the practical process systems do not show a measurable abnormality at the instant of the introduction of the fault. Therefore, there is an inherent delay in fault detection.

Table2- 4: Continuous process variables of TE chemical process (Amin, Imtiaz and Khan, 2018)

Variable Number	Description	Unit
XMEAS (1)	A Feed (Stream 1)	kscmh
XMEAS (2)	D Feed (Stream 2)	kg/hr
XMEAS (3)	E Feed (Stream 3)	kg/hr
XMEAS (4)	A and C Feed (Stream 4)	kscmh
XMEAS (5)	Recycle flow (Stream 8)	kscmh
XMEAS (6)	Reactor feed rate (Stream 6)	kscmh
XMEAS (7)	Reactor pressure	kPa gauge
XMEAS (8)	Reactor level	%
XMEAS (9)	Reactor temperature	° C
XMEAS (10)	Purge rate (Stream 9)	kscmh
XMEAS (11)	Product separator temperature	° C
XMEAS (12)	Product separator level	%
XMEAS (13)	Product separator pressure	kPa gauge
XMEAS (14)	Product separator underflow (Stream 10)	m ³ /hr
XMEAS (15)	Stripper level	%
XMEAS (16)	Stripper pressure	kPa gauge
XMEAS (17)	Stripper underflow (Stream 11)	m ³ /hr
XMEAS (18)	Stripper temperature	° C
XMEAS (19)	Stripper steam flow	kg/hr
XMEAS (20)	Compressor work	kW
XMEAS (21)	Reactor cooling water outlet temperature	° C
XMEAS (22)	Separator cooling water outlet temperature	° C

Table 2- 5: Faults in the TE process and their respective root causes
 (Rato and Reis, 2013), (Amin, Imtiaz and Khan, 2018)

Fault	Description	Type	Root Cause
A	E Feed Loss	Step	XMEAS (3)
B	Reactor cooling water inlet temperature	Random variation	XMEAS (9)
C	Condenser cooling water inlet	Random variation	XMEAS (11)
D	Reactor cooling water valve	Sticking	XMEAS (9)
E	Condenser cooling water valve	Sticking	XMEAS (11)
F	A/C feed ratio, B composition constant	Step	XMEAS (4)
G	Reactor cooling water inlet temperature	Step	XMEAS (9)
H	Condenser cooling water inlet	Step	XMEAS (11)
I	A feed loss (Stream 1)	Step	XMEAS (1)
J	Stripper steam valve stiction	Sticking	XMEAS (19)

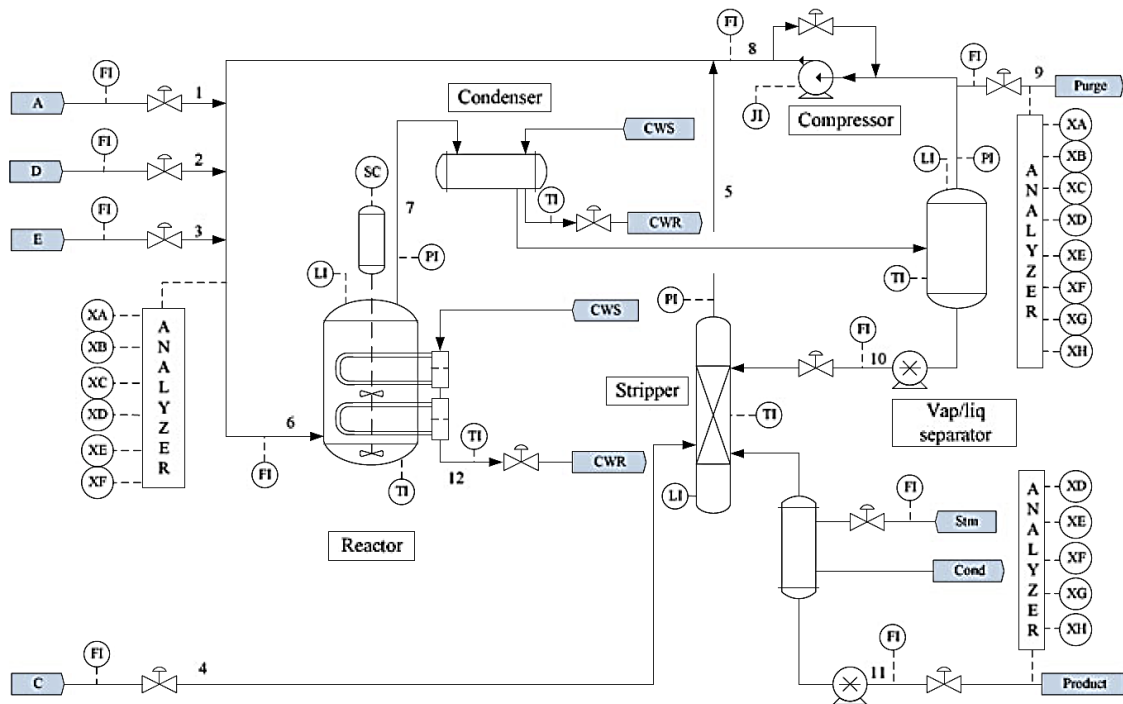


Figure 2-12: Process Flow Diagram of the TE process (Ding, 2014)

2.5.3 Bayesian Network Training

First, an HMM is trained using the fault-free data. Then the first half of the test data were analyzed through HMM to get the detected data strings and their respective log-likelihood values. From this generated data library, the different combinations of states and their respective probabilities can be estimated. The log-likelihood values can be taken as the respective loading of the detected data strings by HMM. The loading value needs to be normalized to be used as conditional probabilities in CPT tables. By establishing the CPTs, and the prior probabilities, BN gains the ability to detect the process abnormalities. The successfully implemented BN is shown in *Figure 2-13*.

2.6 Results and Discussion

2.6.1 Fault detection step

As shown in *Figure 2-14*, the '*findchangepts*' function in Matlab indicates the abnormality and the point where it is detected. From *Table 2-5*, it is clear that XMEAS_3 gives the maximum change in the probability of failure. Therefore XMEAS_3 is considered as the root cause of fault A. Rest of the faults were also analyzed by employing the same methodology. Results are listed in *Table 2-6*. Other than fault F and J, rest of the eight faults are successfully detected and diagnosed by the HMM-BN hybrid approach. There are three probable reasons for the root cause diagnosis inaccuracy of faults F and J. Firstly it can be a weakness of the CPTs. Secondly, inaccuracy of the calculated prior probabilities, and lastly, inaccuracy of the generated likelihood evidence.

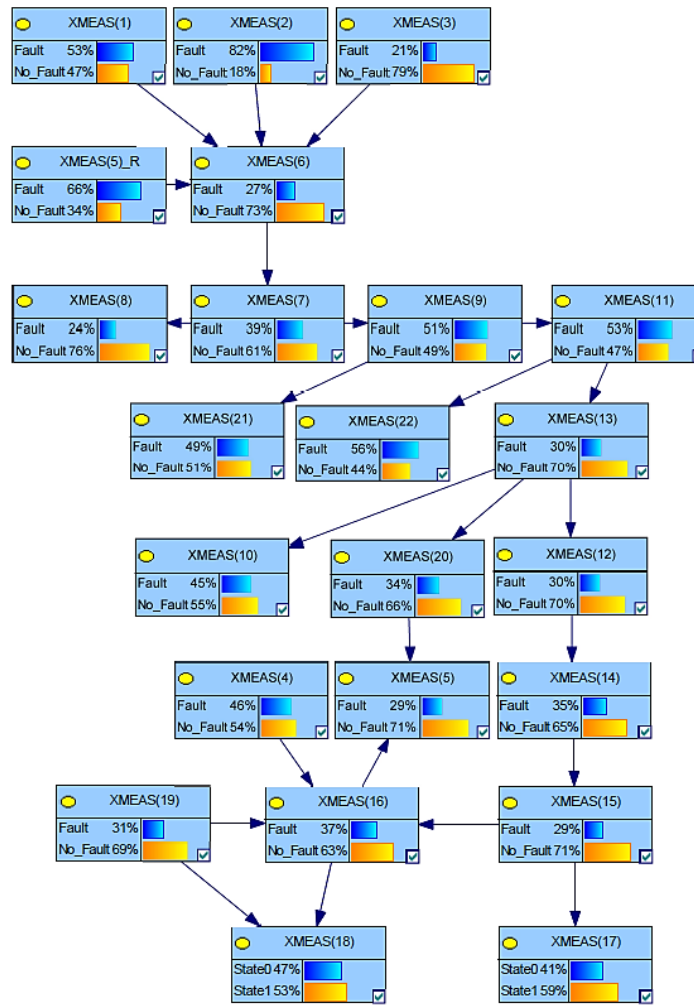


Figure 2-13: BN developed for TE process

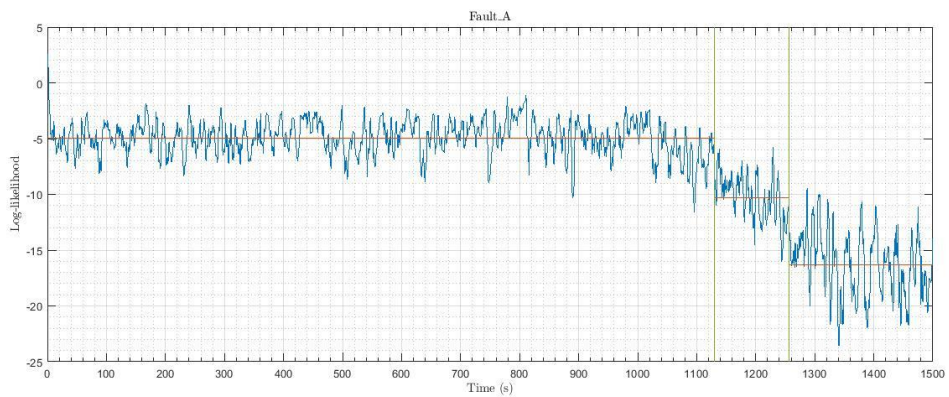


Figure 2-14: Detection of Fault A with HMM trained with NOC data

Table 2-6: Detection times of All ten faults

Fault ID	Time of Detection
A	1125
B	1335
C	1325
D	1335
E	1090
F	1092
G	1325
H	1090
I	1241
J	1139

Table 2-7: Diagnosis of fault A using BN

Name of the Node	IFB	FFB	Difference	Fault Ind
XMEAS_1	53	54	1	✗
XMEAS_2	18	74	5	✓
XMEAS_3	21	82	61	✓
XMEAS_4	46	52		✗
XMEAS_5_R	27	64	37	!
XMEAS_6	26	58		!
XMEAS_7	39	64	25	!
XMEAS_8	24	58		!
XMEAS_9	51	54	3	✗
XMEAS_10	45	48		✗
XMEAS_11	53	68	15	✗
XMEAS_12	30	58		!
XMEAS_13	30	66	36	!
XMEAS_14	35	62		!
XMEAS_15	29	44	15	✗
XMEAS_16	37	68		!
XMEAS_17	41	62	21	!
XMEAS_18	47	42		✗
XMEAS_19	31	62	31	!
XMEAS_20	34	54		✗
XMEAS_21	49	48	1	✗
XMEAS_22	19	58		!

Table2- 8: Performance Comparison of the proposed technique

Fault ID	Original Fault ID	Root Cause	Accurate Diagnosis			
			PCA-T ²	PCA-T ² -BN (Hard Evidence)	PCA-T ² -BN (Multiple likelihood evidence)	HMM-BN (Current Study)
A	E Feed Loss	XMEAS (3)	YES	YES	YES	YES
B	IDV 11	XMEAS (9)	NO	YES	YES	YES
C	IDV 12	XMEAS	NO	YES	YES	YES
D	IDV 14	XMEAS (9)	YES	NO	YES	YES
E	IDV 15	XMEAS	NO	YES	YES	YES
F	IDV 1	XMEAS (4)	NO	YES	YES	NO
G	IDV 4	XMEAS (9)	YES	NO	YES	YES
H	IDV 5	XMEAS	NO	YES	YES	YES
I	IDV 6	XMEAS (1)	YES	YES	YES	YES
J	Stripper steam valve stiction	XMEAS (19)	YES	YES	YES	NO

As shown in *Figure 2-11*, there are three important outputs of HMM namely, time of detection; data string at the time of detection; and LL at the time of detection. One can argue that the data string at the time of detection can be converted into a sequence of 'F,' and 'N' and the LL can be converted into the respective weight (i.e. probability) and send to the BN as likelihood evidence. It was also tested and found less accurate due to the uncertainty of the data sequence. Therefore, once the fault is detected, the Matlab code instantly scans the data strings in the recent past and calculate the probability of failure of all the variables. In other words, the probability of giving a sequence of $F F F \dots F$. The respective probabilities are then used to update the BN with likelihood evidence.

Also, it should be noted that the BN in *Figure 2-13* has two XMEAS (5) nodes, one as a child node and the other one as a parent node. This is because XMEAS (5) is a cyclic variable in TE chemical process. As a BN is acyclic, and it is still required to capture this

cyclic nature into BN, a duplicate dummy node of the recycle flow, XMEAS (5) R, has been created.

2.6.2 A method to simplify the Matlab coding task

The calculation of conditional probabilities using Matlab code can be a tedious task if the correct pattern is not identified. In a complicated Bayesian Network, it will save time by following the procedure below.

Follow a sequence in naming the different nodes in the Bayesian Network (Eg: XMEAS (1), XMEAS (2)). Next, identify and cluster the nodes which are similar in the number of parent nodes (i.e. Nodes with one parent node, two-parent nodes and higher). First, do the coding for a cluster with the highest number of incoming arrows. A simple modification can easily generate the rest of the codes following the pattern. Find and replace option can be effectively used in this step. All possible conditions can be listed in the sequence of binary numbers. For example, if there is a node with two incoming arrows, that means there are four different combinations present. (i.e. 2^2). So 'Faulty' and 'No-Fault' state sequence can be presented as follows. This can be extended with the use of MS Excel with a minimum effort. These generated sequences can be used directly as variables in Matlab code. When exporting the conditional probabilities to 'GeNIe,' the user can directly copy and paste the values to CPT without doing a single manual typing. In brief, the CPT preparation will not be a tedious task if a systematic approach is made. Find and Replace, Copy and Paste options can be used effectively. The entire code can be accessed through <https://github.com/mihiranpathmika/CPL1.0>.

Table 2-9: Sequences for a node with two parent nodes

F	F
F	N
N	F
N	N

2.7 Conclusions

A novel hybrid methodology HMM-BN is proposed in process FDD. The HMM has been used at the first stage to detect the fault using process historical data. A BN is employed to determine a precise diagnosis by reviewing the detection made by HMM. The BN uses the process knowledge and inputs from HMM to perform the diagnosis. Higher diagnostic accuracy is achieved in comparison with PCA-T² approach. In addition to that, fault ‘D’ and ‘G’ are accurately diagnosed which were not successfully diagnosed by PCA-T²-BN hard evidence approach. Due to the inherent fluctuations in LL and practical aspects such as sensitivity of the devices, the HMM does not detect the fault instantly. However, the accuracy of diagnosis demonstrates the strength of new methodology.

The present study has contributed by introducing a new HMM-BN combined approach enhances the FDD capacity of HMM; a novel method to extract information from NOC data, through HMM, to establish the CPTs of BN; a unique and precise way to extract likelihood evidences from HMM to update the BN; and a detailed procedure with Matlab codes of how to adopt the current work to any process system for FDD purposes.

The unique aspect of this study includes: integration of data-driven and knowledge-based methods; easy to implement as a software based on the HMM toolbox Murphy (2005); computationally inexpensive; has the potential to be used as a real-time FDD application

and more importantly does not pre-process the data so useful hidden details are also not filtered.

This work could further be advanced by implementing the developed methodology into software which can detect and diagnose the fault real-time in an industrial application. The Bayes Net Toolbox by Kevin Murphy (Murphy, 2005); can be utilized to do useful computations in the Matlab itself without exporting the data to a separate software such as GeNIe. Also, the delay in detection can be further reduced by optimizing the code. The proposed methodology can be used to predict and prognosis of faults in process systems.

References

- Amin, M. T., Imtiaz, S. and Khan, F. (2018) 'Process system fault detection and diagnosis using a hybrid technique', *Chemical Engineering Science*. Elsevier Ltd, 189, pp. 191–211. doi: 10.1016/j.ces.2018.05.045.
- Baek, S. and Kim, D. Y. (2018) 'Fault Prediction via Symptom Pattern Extraction Using the Discretized State Vectors of Multi-Sensor Signals', *IEEE Transactions on Industrial Informatics*, 3203(c). doi: 10.1109/TII.2018.2828856.
- Bekat, T. *et al.* (2012) 'Prediction of the bottom ash formed in a coal-fired power plant using artificial neural networks', *Energy*. Elsevier, 45(1), pp. 882–887. doi: 10.1016/j.energy.2012.06.075.
- Boyras, P., Acar, M. and Kerr, D. (2007) 'Signal Modelling and Hidden Markov Models for Driving Manoeuvre Recognition and Driver Fault Diagnosis in an urban road scenario', *IEEE Intelligent Vehicles Symposium*, pp. 987–992. doi: 10.1109/IVS.2007.4290245.
- Cai, B., Huang, L. and Xie, M. (2017) 'Bayesian Networks in Fault Diagnosis', *IEEE Transactions on Industrial Informatics*, 13(5), pp. 2227–2240. doi: 10.1109/TII.2017.2695583.

- Cao, Y. *et al.* (2015) ‘Adaptive hidden Markov model with anomaly states for price manipulation detection’, *IEEE Transactions on Neural Networks and Learning Systems*, 26(2), pp. 318–330. doi: 10.1109/TNNLS.2014.2315042.
- Chan, A. and McNaught, K. R. (2008) ‘Using Bayesian networks to improve fault diagnosis during manufacturing tests of mobile telephone infrastructure’, in *Journal of the Operational Research Society*. doi: 10.1057/palgrave.jors.2602388.
- Chelly, S. M. and Denis, C. (2001a) ‘Applying Unsupervised Learning’, *Medicine and science in sports and exercise*, 33(2), pp. 326–333. doi: 10.1111/j.2041-210X.2010.00056.x.
- Chelly, S. M. and Denis, C. (2001b) ‘Applying Unsupervised Learning’, *Medicine and science in sports and exercise*, 33(2), pp. 326–333. doi: 10.1111/j.2041-210X.2010.00056.x.
- Chen, M. Z., Zhou, D. H. and Liu, G. P. (2005) ‘A new particle predictor for fault prediction of nonlinear time-varying systems’, *Developments in Chemical Engineering and Mineral Processing*, 13(3–4).
- Chen, Z. and Yang, Y. (2012) ‘Fault Diagnostics of Helicopter Gearboxes Based on Multi-Sensor Mixture Hidden Markov Models’, *Journal of Vibration and Acoustics*, 134(3), p. 031010. doi: 10.1115/1.4005830.
- Cheng, F., Qu, L. and Qiao, W. (2017) ‘Fault Prognosis and Remaining Useful Life Prediction of Wind Turbine Gearboxes Using Current Signal Analysis’, *IEEE Transactions on Sustainable Energy*, PP(99), p. 1. doi: 10.1109/TSTE.2017.2719626.
- Cheon, S.-P. *et al.* (2009) ‘Bayesian networks based rare event prediction with sensor data’, *Knowledge-Based Systems*. doi: 10.1016/j.knosys.2009.02.004.
- Choi, C. K. and Yoo, H. H. (2014) ‘Precision fault diagnosis procedure for a structural system having a defect employing Hidden Markov Models’, *International Journal of Precision Engineering and Manufacturing*, 15(8), pp. 1667–1673. doi: 10.1007/s12541-014-0517-4.
- Cózar, J., Puerta, J. M. and Gámez, J. A. (2017) ‘An application of dynamic Bayesian

- networks to condition monitoring and fault prediction in a sensed system: A case study', *International Journal of Computational Intelligence Systems*, 10(1), pp. 176–195.
- de Almeida, G. M. and Park, S. W. (2008) *Fault Detection and Diagnosis in the DAMADICS Benchmark Actuator System – A Hidden Markov Model Approach*, *IFAC Proceedings Volumes*. IFAC. doi: 10.3182/20080706-5-KR-1001.02102.
- Deng, Z. (2018) *Proceedings of 2017 Chinese Intelligent Systems Conference*. doi: 10.1007/978-981-10-6499-9.
- Di, Y. *et al.* (2018) 'Fault prediction of power electronics modules and systems under complex working conditions', *Computers in Industry*. Elsevier B.V., 97, pp. 1–9. doi: 10.1016/j.compind.2018.01.011.
- Ding, B. and Fang, H. (2017) 'Fault prediction for nonlinear stochastic system with incipient faults based on particle filter and nonlinear regression', *ISA Transactions*. Elsevier Ltd, 68, pp. 327–334. doi: 10.1016/j.isatra.2017.03.018.
- Ding, S. X. (2014) *Data-driven Design of Fault Diagnosis and Fault-tolerant Control Systems*, *Springer*. doi: 10.1007/978-1-4471-6410-4.
- Downs, J. J. and Vogel, E. F. (1993) 'A plant-wide industrial process control problem', *Computers and Chemical Engineering*. doi: 10.1016/0098-1354(93)80018-I.
- Du, Y. and Du, D. (2018) 'Fault Detection using Empirical Mode Decomposition based PCA and CUSUM with Application to the Tennessee Eastman Process', *IFAC-PapersOnLine*, 51(18), pp. 488–493. doi: 10.1016/j.ifacol.2018.09.377.
- Fulufhelo V. Nelwamondo, T. M. and U. M. (2006) 'Early Classifications of Bearing Faults Using Hidden', *Information and Control*, 2(6), pp. 1281–1299.
- Gao, Q. and Liu, W. (2017) 'Research and Application of the Distillation Column Process Fault Prediction based on the Improved KPCA', pp. 247–251.
- Gharahbagheri, H., Imtiaz, S. A. and Khan, F. (2017) 'Root Cause Diagnosis of Process Fault Using KPCA and Bayesian Network', *Industrial and Engineering Chemistry Research*, 56(8), pp. 2054–2070. doi: 10.1021/acs.iecr.6b01916.
- Gupta, A. and Dhingra, B. (2012) 'Stock market prediction using Hidden Markov Models',

- 2012 Students Conference on Engineering and Systems, SCES 2012. doi: 10.1109/SCES.2012.6199099.
- Hamerly, G. and Elkan, C. (2005a) 'Genetic Programming Approach for Fault Modeling of Electronic Hardware', 2(1), pp. 1563–1569. doi: doi:10.1109/CEC.2005.1554875.
- Hamerly, G. and Elkan, C. (2005b) 'Genetic Programming Approach for Fault Modeling of Electronic Hardware', 2(1), pp. 1563–1569. doi: doi:10.1109/CEC.2005.1554875.
- Hassan, M. R. (2009) 'A combination of hidden Markov model and fuzzy model for stock market forecasting', *Neurocomputing*. Elsevier, 72(16–18), pp. 3439–3446. doi: 10.1016/j.neucom.2008.09.029.
- Hassan, M. R. and Nath, B. (2005) 'Stock market forecasting using hidden Markov model: a new approach', *5th International Conference on Intelligent Systems Design and Applications (ISDA '05)*, pp. 192–196. doi: 10.1109/ISDA.2005.85.
- Hu, Y. *et al.* (2017) 'Dynamic process fault prediction using canonical variable trend analysis', *2017 Chinese Automation Congress (CAC)*, pp. 2015–2020. doi: 10.1109/CAC.2017.8243102.
- Huang, J. H. J. and Zhang, P. Z. P. (2009) 'Fault Diagnosis for Diesel Engines Based on Discrete Hidden Markov Model', *2009 Second International Conference on Intelligent Computation Technology and Automation*, 2, pp. 4–7. doi: 10.1109/ICICTA.2009.358.
- Hughes, G. F. *et al.* (2002) 'Improved disk-drive failure warnings', *IEEE Transactions on Reliability*, 51(3), pp. 350–357. doi: 10.1109/TR.2002.802886.
- Illias, H. A. *et al.* (2015) 'Transformer incipient fault prediction using combined artificial neural network and various particle swarm optimisation techniques', *PLoS ONE*, 10(6), pp. 1–16. doi: 10.1371/journal.pone.0129363.
- Jahani, K. and Razavi, J. (2016) 'Application of Computer Aided Engineering Tools in Performance Prediction and Fault Detection of Mechanical Equipment of Mining Process Line', 10(7), pp. 1295–1299.

- Jia, Y., Sun, L. and Teng, H. (2012) ‘A comparison study of hidden Markov model and particle filtering method: Application to fault diagnosis for gearbox’, *Proceedings of the IEEE 2012 Prognostics and System Health Management Conference (PHM-2012 Beijing)*, pp. 1–7. doi: 10.1109/PHM.2012.6228865.
- Jiang, B.-L., Wey, C.-L. and Fan, L.-J. (1988) ‘FAULT PREDICTION FOR ANALOG CIRCUITS *’, 7(1).
- Jin, S. *et al.* (2017) ‘Bayesian network approach for ceramic shell deformation fault diagnosis in the investment casting process’, *International Journal of Advanced Manufacturing Technology*, 88(1–4), pp. 663–674. doi: 10.1007/s00170-016-8795-x.
- Kang, J. S., Zhang, X. H. and Wang, Y. J. (2011) ‘Continuous hidden Markov model based gear fault diagnosis and incipient fault detection’, *ICQR2MSE 2011 - Proceedings of 2011 International Conference on Quality, Reliability, Risk, Maintenance, and Safety Engineering*, pp. 486–491. doi: 10.1109/ICQR2MSE.2011.5976659.
- Kevin Murphy (2001) *How to use the Bayes Net Toolbox*. Available at: <http://www.cs.ubc.ca/~murphyk/Bayes/usage.11dec01.html> (Accessed: 3 September 2018).
- Kim, J., Han, Y. and Lee, J. (2016) ‘Euclidean Distance Based Feature Selection for Fault Detection Prediction Model in Semiconductor Manufacturing Process’, 133, pp. 85–89.
- Kim, M. J. *et al.* (2011) ‘Optimal Bayesian fault prediction scheme for a partially observable system subject to random failure’, *European Journal of Operational Research*. Elsevier B.V., 214(2), pp. 331–339. doi: 10.1016/j.ejor.2011.04.023.
- Lee, Y. K. *et al.* (2010) ‘A Fault Diagnosis Method for Industrial Gas Turbines Using Bayesian Data Analysis’, *Journal of Engineering for Gas Turbines and Power*, 132(4), p. 041602. doi: 10.1115/1.3204508.
- Li, C. *et al.* (2017) ‘Fault diagnosis and prediction of complex system based on Hidden Markov model’, *Journal of Intelligent & Fuzzy Systems*, 33(5), pp. 2937–2944. doi: 10.3233/JIFS-169344.

- Li, L. *et al.* (2014) ‘Trend prognosis of aero-engine abrupt failure based on affinity propagation clustering’, in Jinsong, W. (ed.) *In Proceedings of the First Symposium on Aviation Maintenance and Management*. Berlin, Heidelberg.: Springer, pp. 13–22.
- Li, Z. *et al.* (2005) ‘Hidden Markov model-based fault diagnostics method in speed-up and speed-down process for rotating machinery’, *Mechanical Systems and Signal Processing*, 19(2), pp. 329–339. doi: 10.1016/j.ymsp.2004.01.001.
- Li, Z. *et al.* (2006) ‘Fault recognition method for speed-up and speed-down process of rotating machinery based on independent component analysis and Factorial Hidden Markov Model’, *Journal of Sound and Vibration*, 291(1–2), pp. 60–71. doi: 10.1016/j.jsv.2005.05.020.
- Li, Z., Fang, H. and Xia, L. (2014) ‘Increasing mapping based hidden Markov model for dynamic process monitoring and diagnosis’, *Expert Systems with Applications*. Elsevier Ltd, 41(2), pp. 744–751. doi: 10.1016/j.eswa.2013.07.098.
- Liu, K., Zhang, X. and Shi, J. (2014) ‘Adaptive Sensor Allocation Strategy for Process Monitoring and Diagnosis in a Bayesian Network.’, *IEEE Transactions on Automation Science & Engineering*, 11(2), pp. 452–462. Available at: 10.1109/TASE.2013.2287101%5Cnhttp://search.ebscohost.com/login.aspx?direct=true&db=bth&AN=95633411&site=ehost-live.
- Ma, J. and Xu, J. (2015) ‘Fault Prediction Algorithm for Multiple Mode Process Based on Reconstruction Technique’, 2015. doi: 10.1155/2015/348729.
- Mallick, M. R. (2013) *Hybrid Method for Process Fault Detection and Diagnosis*. Memorial University of Newfoundland, Canada.
- Mu, F. and Venkatasubramanian, V. (2003) ‘Online HAZOP analysis for abnormal event management of batch process’, *Computer Aided Chemical Engineering*, 14(C), pp. 803–808. doi: 10.1016/S1570-7946(03)80215-8.
- Murphy, K. (2005) *Hidden Markov Model (HMM) Toolbox for Matlab, Distributed under the MIT License*. Available at: <http://www.cs.ubc.ca/~murphyk/Software/HMM/hmm.html> (Accessed: 19 July

2018).

- Murray, J. (2003) 'Hard drive failure prediction using non-parametric statistical methods', *Proceedings of ICANN/ ...*, (1). Available at: <http://hebb.mit.edu/people/jfmurray/publications/Murray2003.pdf>.
- Neapolitan, R. E. (2010) 'Learning Bayesian Networks', *International journal of data mining and bioinformatics*, 4, pp. 505–19. doi: 10.1016/j.jbi.2010.03.005.
- Nelwamondo, F. V, Marwala, T. and Mahola, U. (2006) 'Early classifications of bearing faults using hidden Markov models, Gaussian mixture models, Mel frequency cepstral coefficients and fractals', *International Journal of Innovative Computing, Information and Control*, 2(6), pp. 1281–1299.
- Nguyen, N. (2016) 'Stock Price Prediction using Hidden Markov Model', pp. 1–20. Available at: https://editorialexpress.com/cgi-bin/conference/download.cgi?db_name=SILC2016&paper_id=38.
- Onel, M. *et al.* (2018) 'Big data approach to batch process monitoring: Simultaneous fault detection and diagnosis using nonlinear support vector machine-based feature selection', *Computers & Chemical Engineering*. Elsevier Ltd, 115, pp. 46–63. doi: 10.1016/j.compchemeng.2018.03.025.
- Pathmika, M. and Khan, F. (2018a) *Dynamic process fault detection and diagnosis based on a combined approach of hidden Markov and Bayesian network model*. St. John's.
- Pathmika, M. and Khan, F. (2018b) *Process fault prediction and prognosis using a hybrid model*. St. John's.
- Predix (2016a) *Machine Learning*. Available at: <https://www.predix.io/analytics>.
- Predix (2016b) *Machine Learning*.
- Qi, F. and Huang, B. (2011) 'Bayesian methods for control loop diagnosis in the presence of temporal dependent evidences', *Automatica*. Elsevier Ltd, 47(7), pp. 1349–1356. doi: 10.1016/j.automatica.2011.02.015.
- Rabiner, L. and Juang, B. (1986) 'An introduction to hidden Markov models', *IEEE ASSP Magazine*, 3(January), p. Appendix 3A. doi: 10.1109/MASSP.1986.1165342.
- Rabiner, L. R. R. (1989) 'Tutorial on Hidden Markov Models and Selected Applications

- in Speech Recognition', *Ieee*, 77(2), pp. p257-286. doi: 10.1109/5.18626.
- Ragab, A. *et al.* (2018) 'Fault diagnosis in industrial chemical processes using interpretable patterns based on Logical Analysis of Data', *Expert Systems with Applications*. doi: 10.1016/j.eswa.2017.11.045.
- Ramana, E. V., Sapthagiri, S. and Srinivas, P. (2017) 'Data Mining Approach for Quality Prediction and Fault Diagnosis of Injection Molding Process', *Indian Journal of Science and Technology*, 10(17), pp. 1–7. doi: 10.17485/ijst/2017/v10i17/112580.
- Rato, T. J. and Reis, M. S. (2013) 'Fault detection in the Tennessee Eastman benchmark process using dynamic principal components analysis based on decorrelated residuals (DPCA-DR)', *Chemometrics and Intelligent Laboratory Systems*. Elsevier B.V., 125, pp. 101–108. doi: 10.1016/j.chemolab.2013.04.002.
- Ren, X. *et al.* (2017) 'Fault Detection and Diagnosis for Nonlinear and Non-Gaussian Processes Based on Copula Subspace Division', *Industrial and Engineering Chemistry Research*. doi: 10.1021/acs.iecr.7b02419.
- Ruggeri, F., Faltin, F. and Kenett, R. (2007) 'Bayesian Networks', *Encyclopedia of Statistics in Quality & Reliability*, 1(1), p. 4. doi: 10.1002/wics.48.
- Sadeghian, M. and Fatehi, A. (2011) 'Identification, prediction and detection of the process fault in a cement rotary kiln by locally linear neuro-fuzzy technique', *Journal of Process Control*, 21(2), pp. 302–308. doi: 10.1016/j.jprocont.2010.10.009.
- Sadhu, A., Prakash, G. and Narasimhan, S. (2017) 'A hybrid hidden Markov model towards fault detection of rotating components', *JVC/Journal of Vibration and Control*, 23(19), pp. 3175–3195. doi: 10.1177/1077546315627934.
- Santos, E. B. *et al.* (2014) 'Bayesian classifiers applied to the Tennessee Eastman process', *Risk Analysis*, 34(3), pp. 485–497. doi: 10.1111/risa.12112.
- Sawaragi, Y., Soeda, T. and Omatu, S. (1978) *{L}ecture {N}otes in {C}ontrol and {I}nformation {S}ciences*. 1st edn. Springer.
- Soualhi, a. *et al.* (2012) 'Fault detection and diagnosis of induction motors based on hidden Markov model', *2012 XXth International Conference on Electrical Machines*, pp. 1693–1699. doi: 10.1109/ICEIMach.2012.6350108.

- The Mathworks (2016) *Introducing Machine Learning What is Machine Learning, Perspectives on Ontology Learning*.
- Tobon-Mejia, D. A. *et al.* (2010) ‘A Mixture of Gaussians Hidden Markov Model for failure diagnostic and prognostic’, *2010 IEEE International Conference on Automation Science and Engineering, CASE 2010*, (May 2014), pp. 338–343. doi: 10.1109/COASE.2010.5584759.
- Ündey, C., Tatara, E. and Çinar, A. (2004) ‘Intelligent real-time performance monitoring and quality prediction for batch/fed-batch cultivations’, *Journal of Biotechnology*, 108(1), pp. 61–77. doi: 10.1016/j.jbiotec.2003.10.004.
- Venkat, V. *et al.* (2003) ‘A review of process fault detection and diagnosis: Part III: Process history based methods’, *Computers & Chemical Engineering*, 27(3), pp. 327–346. doi: 10.1016/S0098-1354(02)00162-X.
- Venkatasubramanian, V., Rengaswamy, R. and Kavuri, S. N. (2003) ‘A review of process fault detection and diagnosis: Part II: Qualitative models and search strategies’, *Computers & Chemical Engineering*, 27(3), pp. 313–326. doi: 10.1016/S0098-1354(02)00161-8.
- Verron, S., Tiplica, T. and Kobi, A. (2010) ‘Fault diagnosis of industrial systems by conditional Gaussian network including a distance rejection criterion’, *Engineering Applications of Artificial Intelligence*. Elsevier, 23(7), pp. 1229–1235. doi: 10.1016/j.engappai.2010.05.002.
- Wang, D. *et al.* (2009) ‘“Bearing Fault Diagnosis Using Singular Value Decomposition and Hidden Markov Modeling”’, *ASME 2009 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, pp. 1–7.
- Wang, F. *et al.* (2016) ‘Hidden Markov Model-Based Fault Detection Approach for a Multimode Process’, *Industrial and Engineering Chemistry Research*, 55(16), pp. 4613–4621. doi: 10.1021/acs.iecr.5b04777.
- Wang, H. Q. *et al.* (2018) ‘Data-driven fault prediction and anomaly measurement for complex systems using support vector probability density estimation’, *Engineering*

- Applications of Artificial Intelligence*. Elsevier Ltd, 67(December 2016), pp. 1–13. doi: 10.1016/j.engappai.2017.09.008.
- Wang, L. *et al.* (2015) ‘Switched fault diagnosis approach for industrial processes based on hidden Markov model’, *European Workshop on Advanced Control and Diagnosis*, pp. 1–11. doi: 10.1088/1742-6596/659/1/012047.
- Wong, W. C. and Lee, J. H. (2009) ‘Fault Detection in Process Systems using Hidden Markov Disturbance Models’, *IFAC Proceedings Volumes*, 42(11), pp. 291–296. doi: 10.3182/20090712-4-TR-2008.00045.
- Wu, J. *et al.* (2017) ‘Probabilistic analysis of natural gas pipeline network accident based on Bayesian network’, *Journal of Loss Prevention in the Process Industries*. Elsevier Ltd, 46, pp. 126–136. doi: 10.1016/j.jlp.2017.01.025.
- Xiao, W. (2016) ‘A Probabilistic Machine Learning Approach to Detect Industrial Plant Faults’, pp. 1–24. Available at: <http://arxiv.org/abs/1603.05770>.
- Xie, S. (2013) ‘Fault diagnosis of the satellite power system based on the Bayesian network’, in *Computer Science & Education (ICCSE), 2013 8th International Conference on*. IEEE, pp. 1004--1008.
- Yan, R.-W. and Cai, J.-D. (2009) ‘Application of Hidden Markov Model to Fault Diagnosis of Power Electronic Circuit’, *2009 IEEE Circuits and Systems International Conference on Testing and Diagnosis*, pp. 1–4. doi: 10.1109/CAS-ICTD.2009.4960787.
- Yu, H. *et al.* (2014) ‘Self-Organizing Map Based Fault Diagnosis Technique for Non-Gaussian Processes’, *Industrial & Engineering Chemistry Research*, 53(21), pp. 8831–8843. doi: 10.1021/ie500815a.
- Yu, H., Khan, F. and Garaniya, V. (2015) ‘Modified Independent Component Analysis and Bayesian Network- Based Two-Stage Fault Diagnosis of Process Operations Modified Independent Component Analysis and Bayesian Network- Based Two-Stage Fault Diagnosis of Process Operations’, (March). doi: 10.1021/ie503530v.
- Yu, L., Wang, S. and Lai, K. K. (2005) ‘A novel nonlinear ensemble forecasting model incorporating GLAR and ANN for foreign exchange rates’, *Computers and*

- Operations Research*, 32(10), pp. 2523–2541. doi: 10.1016/j.cor.2004.06.024.
- Yusuf, S. *et al.* (2013) ‘Fault classification improvement in industrial condition monitoring via hidden markov models and naïve bayesian modeling’, *ISIEA 2013 - 2013 IEEE Symposium on Industrial Electronics and Applications*, pp. 75–80. doi: 10.1109/ISIEA.2013.6738971.
- Yuwono, M. *et al.* (2016) ‘Automatic bearing fault diagnosis using particle swarm clustering and Hidden Markov Model’, *Engineering Applications of Artificial Intelligence*. Elsevier, 47, pp. 88–100. doi: 10.1016/j.engappai.2015.03.007.
- Zadakbar, O., Imtiaz, S. and Khan, F. (2012) ‘Dynamic Risk Assessment and Fault Detection Using Principal Component Analysis’, *Industrial & Engineering Chemistry Research*, 52, pp. 809–816. doi: 10.1021/ie202880w.
- Zadakbar, O., Imtiaz, S. and Khan, F. (2013) ‘Dynamic Risk Assessment and Fault Detection Using Principal Component Analysis’, *Industrial & Engineering Chemistry Research*, 52(2), pp. 809–816. doi: 10.1021/ie202880w.
- Zhang, S. *et al.* (2017) ‘Data-Based Line Trip Fault Prediction in Power Systems Using LSTM Networks and SVM’, *IEEE Access*, 6, pp. 7675–7686. doi: 10.1109/ACCESS.2017.2785763.
- Zhao, Y. *et al.* (2017) ‘Fault prediction and diagnosis of wind turbine generators using SCADA data’, *Energies*, 10(8), pp. 1–17. doi: 10.3390/en10081210.
- Zhou, H. *et al.* (2016) ‘Bearing fault recognition method based on neighbourhood component analysis and coupled hidden Markov model’, *Mechanical Systems and Signal Processing*. Elsevier, 66–67, pp. 568–581. doi: 10.1016/j.ymsp.2015.04.037.

CHAPTER 3 : PROCESS FAULT PREDICTION AND PROGNOSIS USING A HYBRID MODEL²

Co-authorship Statement

Galagedarage Don Mihiran Pathmika is the principal author of this thesis. He has undertaken the research and prepared the first draft. Professor Faisal Khan, the co-author of the manuscript, shared the problem and conceptualized the methodology. In addition, Prof. Khan contributed by reviewing, and revising the manuscript. He also guided the author throughout the entire process of the methodology development, testing, validation and its application development. The software code and analysis of results were solely contributed by the principal author, and the results were validated for correctness by the co-author.

Abstract

Prediction and management of the process faults could save billions of dollars per year. This study proposes a hybrid approach to predict and prognosis process faults. The hybrid approach is comprised of a Hidden Markov Model (HMM) and Bayesian Network (BN). HMM predicts the abnormalities using process historical data while the BN uses the process knowledge to prognose the fault. In the off-line component, an HMM which is trained with Normal Operating Condition data is used to determine the log-likelihoods (LL) of each process history data string. The generated LL values are then used to develop the Conditional Probability Tables of BN while the structure of BN is established based on

² This chapter is submitted as a manuscript to the journal of Chemical Engineering Research and Design and currently under review

process knowledge and with the help of Sign Directed Graphs. In addition to that, a separate historical data set with known faults is used to generate a database of LL values concerning the same HMM trained with standard operating condition data. In the online component, the trained HMM is used to check the LL values of incoming data strings continuously and compares with the LL historical database. Based on the comparison, the system decides the most likely future condition of the system in n number of seconds. The time of prediction of abnormality and probabilities of being in specific operational state at the predicted time is used to generate the likelihood evidence to BN. The updated BN along with likelihood evidence is then used to prognose the cause. Performance of the proposed approach is tested using published data of Tennessee Eastman Process. The system can predict all the selected ten faults while accurately prognosis eight of them.

Keywords: *Process fault prognosis; Hidden Markov model; Bayesian network, Process fault prediction*

3.1 Introduction and Review of the Relevant Literature

There are many options available in fault prediction using machine learning approaches. This topic discusses a variety of machine learning algorithms can be used in process fault detection and prediction. As illustrated in *Figure 3-1*, machine learning approaches can be divided into two basic categories called, supervised learning and unsupervised learning. In supervised learning, the algorithms assess the input data and corresponding outputs to learn the mapping function from input to the output. In unsupervised learning, the algorithms identify the hidden structures of data for further evaluation of data.

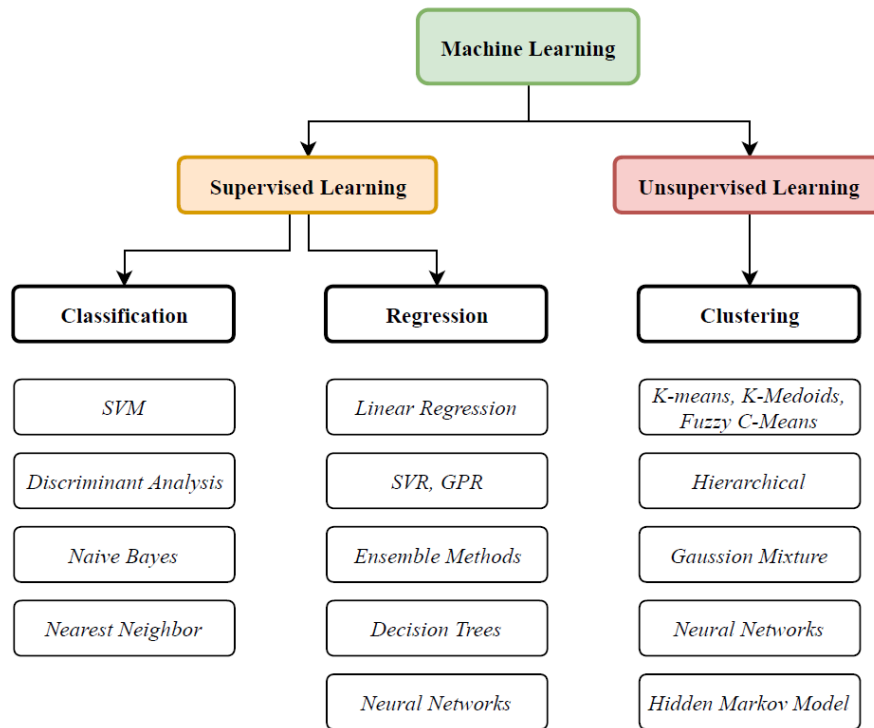


Figure 3-1: Different machine learning techniques (The Mathworks, 2016)

Classification and Regression come under supervised learning while clustering comes under unsupervised learning. Examples for each category are also illustrated in Figure 3-1. (The Mathworks, 2016).

As further mentioned about supervised learning in (The Mathworks, 2016), to predict future outputs, a model needs to be trained based on input and respective output data. It builds a model, with the available known inputs and respective outputs, which can be used to predict the output of an unknown input. In classification, the incoming data can be classified into predefined groups. For example, it can be used to detect whether an e-mail is spam or not. On the other hand, regression techniques can predict continuous changes of

quantities such as changes in temperature or pressure of a polymer melt. They are widely used in electricity load forecasting (The Mathworks, 2016).

Binary classification problems can be adequately handled using logistic regression by using as the first step. It can be used to predict the probability of a binary response by fitting a suitable model. It is more efficient when the data is separable by one linear boundary. Also, k Nearest Neighbor (kNN) is a simple algorithm which is useful to use when the concern on memory usage and prediction speed are less concern. The primary assumption in kNN is that the objects near each other are similar. It can categorize objects based on the classes of their nearest neighbours in the data set. Distance matrices are used to locate the nearest neighbour. Further, Neural networks can be used for modelling highly nonlinear systems. It can also facilitate the constant update of data with the availability. It is a characteristic of a neural network having highly connected networks of neurons that interconnect the inputs with the respective outputs, inspired by the human brain (The Mathworks, 2016).

As further explained in (The Mathworks, 2016), on the other hand, Naïve Bayes classifies incoming data by assessing the probability of belonging to a particular class. It also assumes each class has unique features which do not have any similarities. This technique performs well for a small dataset containing many parameters. The discriminant analysis finds linear combinations of features to classify data. The primary assumption in the discriminant analysis is that Gaussian distribution is the basis for the generation of data for training. If there is a requirement of an easily interpretable simple model, discriminant analysis is a good option. The model created using discriminant analysis are fast to predict, and memory usage can be optimized during the training process. A decision tree is also a

technique that has a minimum memory usage. It does not give a high predictive accuracy but easy to interpret and fast to fit. When the decision trees are lower in performance, several low-performance decision trees are combined to an ensemble called Bagged and Boosted Decision Trees. Each tree is trained independently. This version of decision trees is much suitable when the time taken to train a model is not critical.

It is also mentioned in (The Mathworks, 2016) that linear regression is a technique that is easy to interpret and train. Therefore, it is the first model to fit into a given data set. A continuous response variable can be described using this technique as a linear function of one or more predictor variables. On the other hand, to describe nonlinear relationships, nonlinear regression technique is used. The effectiveness of this technique is evident when the data has a strong nonlinear trends and cannot be quickly transformed into a narrow space. Gaussian process regression (GPR) models are used for predicting the value of a continuous response variable. GPRs are non-parametric models which are widely used in spatial analysis. When there is uncertainty, GPR can make interpolations to make predictions. Support Vector Machine Regression (SVMR) is also has a similar operation to SVM Classification. SVM regression algorithms work similar to that of SVM classification algorithms though they are modified to be able to predict a continuous response. This technique is effectively used in high-dimensional data.

Further, when the response variables have non-normal distributions, Generalized Linear Model can be successfully used which is a particular case of nonlinear models which utilizes linear methods. Similar to decision trees used in classification, the decision trees developed for regression are called Regression Trees. These are specially modified to

predict continuous responses. When the predictors are nonlinear and discrete, this technique can be successfully used (The Mathworks, 2016).

The specialty of unsupervised learning is, it can find hidden patterns or data structures in input data. The most widely used unsupervised learning technique is called Clustering. It is used for exploratory data analysis to uncover hidden patterns or groupings in data. Clustering can again be divided into two basic categories namely; hard, and soft. Regarding hard clustering examples, k-means and k-medoids are closely related to each other. The only difference is the latter does coincide data points, and the former does not. In hierarchical clustering, the data are grouped into a binary hierarchical tree while the self-organizing map is a Neural-network based clustering technique that transforms a data set into a 2D plot. About Fuzzy C-means, which is a soft clustering technique, can be used when data points belong to more than one cluster. This technique is widely used in pattern recognition. Similar to Fuzzy C-means, the Gaussian Mixture Model (GMM) is also a partition-based clustering technique where data points come from different multivariate normal distributions (Chelly and Denis, 2001a).

On top of this, there are three most widely used dimensionality reduction techniques namely: Principal component analysis (PCA); Factor analysis; and Nonnegative matrix factorization. In PCA, few principal components can capture a high dimensional data set by performing a linear transformation on data. The strength of this method is, the principal components can catch most of the variance or information of the entire data set. The relationships between variables in a given data set can be identified by factor analysis and representation regarding a lesser number of latent, or common factors.

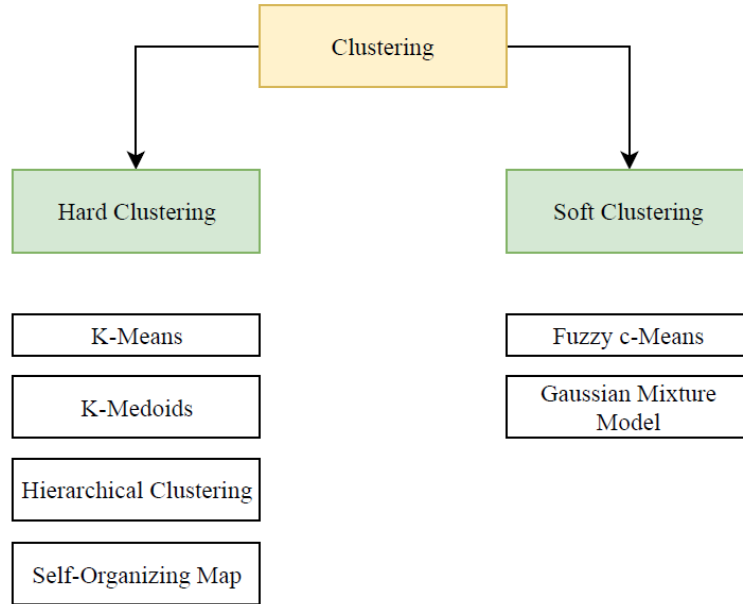


Figure 3-2: Different techniques in Clustering

Nonnegative (\mathbb{R}_0^+) matrix factorization is employed when dealing with non-negative quantities (Chelly and Denis, 2001b).

Machine learning techniques used as prediction tools in a variety of applications are discussed under this topic. There are numerous examples available in the literature for both individual and combined use of machine learning techniques.

A failure prediction methodology of a partially observable system is presented by (Kim *et al.*, 2011). They have modelled the system behaviour three hidden state continuous time-homogeneous Markov process. States 0 and 1 are not observable. Those two states represent normal and warning conditions respectively. Only visible failure state is 2. EM algorithm is employed to model parameters estimation. Further, a cost-optimal Bayesian fault prediction scheme is also applied. A comparison concerning other prediction techniques is given. The Effectiveness of the proposed approach is clearly illustrated.

Logistic Regression Classifier (LRC) is a powerful tool in predicting linearly separable classes. It is a commonly used analytical model for classification problems. When a training feature matrix X is provided along with the corresponding target vector Y , a logistic regression model can be trained to predict Y for even unseen instances of X . The input has two main components namely; data, and parameters. The data component includes the training and predicted datasets. The training dataset requires the feature values and their corresponding target values, while the test dataset only requires the feature values to predict their unknown target values (Predix, 2016a).

As further mentioned in (Predix, 2016b), Random Decision Forests (RDF) is a combined approach of learning methods for tasks such as classification, and regression. Initially, a set of decision trees is generated during training time. As outputs, the class that is the mode of the classes (classification) or mean prediction (regression) of the individual trees can be mentioned. It trains some decision trees from bootstrap samples from the training set with replacement. On the other hand, the algorithm also draws a random subset of features for training the individual trees. This approach makes the trees more independent in comparison with the conventional approach. In the case of classification, the majority rule is used on trained decision trees to classify the new data. While each decision tree comes up with a decision, the next prediction is selected based on the number of votes. Random Forest results provide better performance in prediction. Random Forest is practically tuning free therefore it is not required to do any parameter tuning to find the optimal model.

Two alternatives of Genetic Programming (GP) approaches introduced by (Hamerly and Elkan, 2005a). They can be used in intelligent online performance monitoring of electronic

circuits and systems. A stressor susceptibility interaction model is introduced to assess the reliability of circuit systems. When a stressor exceeds the susceptibility limit, the system is identified as a failure. Direct measurements through sensors are used to generate the validated stressor vectors and then they are fed to the GP model. The results are compared with ANN outputs and found to be useful in performance.

Further, (Hamerly and Elkan, 2005b) propose a technique to predict hard disk failures which are rare but costly. Two Bayesian methods are introduced namely; a mixture model of Naïve Bayes sub-models, and Naïve Bayes Classifier. A former method used the EM algorithm while the latter is a supervised learning approach. Both techniques show good prediction accuracy on real-world data. Also, (Murray, 2003) confirms that the rank-sum method outperforms other techniques in comparison with the performance of support vector machines (SVMs), unsupervised clustering, and non-parametric statistical tests. Two improved versions of Self-Monitoring and Reporting Technology (SMART) failure prediction system are proposed by (Hughes *et al.*, 2002). The proposed techniques give several times higher prediction accuracy than error thresholds on hard disk drives which are prone to fail, at 0.2 % false alarm rate.

Bearing failure prediction was successfully done by (Fulufhelo V. Nelwamondo, 2006) using HMM and GMM. They introduce feature extraction methodologies that can facilitate early detection of faults. Time domain vibration signals of faulty and standard bearings in rotating machinery are used for feature extraction. HMM and GMM is then used to classify faults based on the extracted features. Based on the classification performance, HMM is

superior to that of GMM. They further mention that HMM has a disadvantage of being computationally expensive.

Euclidean distance based feature selection is proposed by (Kim, Han and Lee, 2016). It is used for fault detection and prediction model in the semiconductor synthesizing process. As the first step, the features of the semiconductor manufacturing process are measured regarding Mean Absolute Value (MAV) and Standard deviation (SD). After that, using the Euclidean distance, the most appropriate features are selected using the classification model. Finally, with the filtered features, the neural network is trained to generate a fault prediction model. The proposed method performs well in the semiconductor manufacturing process fault prediction.

Neural network modelling is used by (Bekat *et al.*, 2012) to predict the amount of bottom ash accumulated in a pulverized coal-fired power plant. Operating data collected throughout the one-year period and the properties of the coal processed are used in the prediction process. Network architecture used is Feedforward. Backpropagation learning is used with three layers. The sigmoid function is used as the activation function. The authors have determined the ideal parameters for accurate predictions along with the most useful metrics to be monitored based on a sensitivity analysis.

To minimize the error and reduce the maintenance cost, prediction of an incipient fault in transformer oil is essential. Artificial neural network (ANN), and particle swarm optimization (PSO) based methodology is presented by (Illias *et al.*, 2015) to predict the incipient transformer fault. It has been found that the ANN-PSO method proposed gives the leading percentage of correct identification.

On the other hand, (Jiang, Wey and Fan, 1988) propose an algorithm to predict faults in analog circuits. The central concept is continuously monitoring the component values which are evaluated according to the consecutive voltage measurements. These measurements are taken at the accessible test points, at each periodic maintenance. This approach makes it possible to locate the faulty components as well as components which are predicted to be failed shortly.

Based on multi-PCA model, (Ma and Xu, 2015) present a methodology for multiple mode process fault detection. It also includes techniques for fault estimation and fault prediction. Multi PCA model is initially used to detect faults in a process which is operated under steady state and different conditions. For the transition process, a weighted algorithm is used. Fault amplitude is made consistent by using a consistent estimation algorithm, and finally, SVM is used to predict the fault amplitude changing pattern. This method has a proven performance by applying and testing in Tennessee Eastman process data.

A prediction technique is introduced by (Gao and Liu, 2017), which is developed based on an improved version of kernel principal component analysis (KPCA) method. Indiscernibility and eigenvector concepts are used. The application area is process fault prediction of distillation columns. This version of KPCA can remove variables with almost no correlation to the fault being monitored. On the other hand, it can reduce the number of data strings used several times. Proposed methodology gives better performance over the traditional technique. By applying the method in a distillation column scenario, the authors have shown that the KPCA method is capable of predicting the process failures caused by small disturbance.

Weighted least square vector machines regression is used by (Gao and Liu, 2017) to develop a Hammerstein model to predict the dynamic behaviour and the possible faults in Imperial Smelting Furnace (ISF). The proposed model is capable of accurate fault predictions of ISF. Further, (Ramana, Sathagiri and Srinivas, 2017) have introduced a prediction methodology for the quality of injection moulding products based on a machine learning approach, which has shown a prediction accuracy of 95%. It is an effective method to increase the productivity by eliminating defectives during a production process. To build the data mining models, Decision Tree, and k-NN techniques are used and trained using a training data set developed using actual production data of a given product. Then the prediction accuracy is tested using a testing dataset developed similarly.

The use of Computer Aided Engineering (CAE) tools on fault prediction in Copper processing line is studied by (Jahani and Razavi, 2016). The outcome of the study is highly useful in the predictive maintenance of critical equipment such as slurry pumps and hydro-cyclones. Further, the simulations can be used to make decisions on optimum-measuring parameters, intervals, and their respective locations.

Prediction of emerging faults of dynamic industrial processes is achieved by (Hu *et al.*, 2017), using an approach based on Canonical Variable Trend Analysis (CVTA). Canonical Variable is the leading information carry forward to make the predictions. They are the un-correlated latent features extracted through the analysis of process dynamics. The initial analysis is done Canonical Variate Analysis (CVA) algorithm while SVM is employed to identify the relationship between past and future values. It facilitates the development of a time series prediction model for the canonical variables. Change of the process status is

forecasted, using an overall monitoring statistic and based on the predicted canonical variables. They have demonstrated the effectiveness of the technique by applying it in a simulation on a Continuous Stirred Tank Reactor (CSTR) system.

An intelligent algorithm for fault prediction of turbine pitch system is proposed by (Deng, 2018), based on Least Squares Support Vector Machines (LS-SVM) parameter optimization. Initially, the data of the SCADA system are analyzed. Through this, four kinds of parameters are selected as the input of the model, which are strictly related to the turbine pitch system fault. Then the minimum output coding (MOC) is introduced to construct multiple classifications LS-SVM to understand the multi-class classification of pitch fault. Later, to select the optimal feature parameters for the multi-class LS-SVM classifiers, the algorithm of particle swarm optimization is employed. The proposed methodology is applied to a pitch fault prediction scenario of wind farms. The performance is compared with the neural network algorithm (back propagation) and the standard SVM algorithm. The proposed method is found to be superior in performance.

A method for line trip fault prediction in power systems is proposed by (Zhang *et al.*, 2017). It is done based on long-short-term memory (LSTM) networks and SVM. Further, LSTM networks are used to capture temporal features of multisource data as they perform well in extracting the features of time series for a long-time span. To get the final prediction results, SVM is used for classification. The actual data for experiments is obtained from the Wanjiang substation in the China Southern Power Grid. Improved performance of the proposed combined approach of LSTM and SVM is evident in comparison with the current data mining methods.

A novel approach for power converter fault prediction in power conversion systems is presented by (Di *et al.*, 2018). Decision tree and SVM are used. Those two will take in to account the changes in working conditions and imbalances of data respectively. It was validated with an industrial application to be useful in predicting the power converter failures.

A prediction method for cement rotary kiln process is proposed by (Sadeghian and Fatehi, 2011), using a nonlinear system identification method. First, the suitable inputs and outputs are selected, and a model of inputs and outputs are identified for the complete system. To identify various operating points of the kiln, Locally Linear Neuro-Fuzzy (LLNF) model is used. An incremental tree structure algorithm is employed to train the model. The methodology is used to develop three models, one for normal operating conditions and the other two for two faulty situations. The proposed technique can predict the fault occurrence 7 minutes in advance.

In order to calculate the probability of fault prediction, a method has been proposed by (Chen, Zhou and Liu, 2005). This method can be used for nonlinear time-varying systems. It is a particle predictor-based method. As illustrated by the use of simulations, the proposed methodology is capable of giving an early alarm can be triggered before the system reaches the faulty state. Although the conventional Particle Filter cannot perform with unknown time-varying parameters, the Particle Predictor has the ability. As further explained in (Chen, Zhou and Liu, 2005), it is almost impossible to make a prediction on the abrupt faults of a system. Nevertheless, the slowly developing faults can be predicted with the use of an online monitoring system.

In a nonlinear stochastic system, incipient faults prediction methodology is proposed by (Ding and Fang, 2017). This fault estimation algorithm is developed based on particle filter. Some simplifying assumptions on the incipient faults are made without losing critical details. A novel fault detection strategy called ‘intuitive fault detection’ is presented. When the incipient faults are detected, nonlinear regression is used to identify the respective parameters. Based on the parameters determined, the oncoming fault signal is predicted. Finally, a standard simulation has been employed to verify the performance of the new methodology.

Remaining useful life (RUL) of a wind turbine is an important parameter to know, in order to maintain the reliability of the service. By employing an adaptive neuro-fuzzy inference system (ANFIS) along with particle filtering (PF) approaches, (Cheng, Qu and Qiao, 2017) propose a method for fault prognosis and gearbox RUL prediction. The fault features are extracted from the stator current of the generator coupled with the gearbox. The extracted fault features are used to train ANFIS, and the PF predicts the RUL. For this, new information of the fault features is also used. The proposed method is found effective based on the experimental results. A similar problem is successfully solved by (Zhao *et al.*, 2017). It is shown that RUL of the wind turbine is predictable 18 days ahead, with nearly 80% accuracy. The generator faults are diagnosable with an accuracy of 94% once occurred. The benefit of the proposed system is, it does not require any additional hardware installation. Already available SCADA system serves as the source of information hence cost efficient. With a detailed analysis, the authors have selected the SVM as the most

suitable classification technique for this particular application among ANN, Bayes classifier, k-NN classifiers.

Prediction of the foreign exchange rate is also a profoundly explored researched area. A nonlinear ensemble forecasting model is proposed by (Yu, Wang and Lai, 2005). It is recommended as an alternative tool for exchange rate forecasting. It consists of generalized linear auto-regression (GLAR) with artificial neural networks (ANN) for accurate predictions. Performance of the new combined model is compared with the two individual forecasting tools (i.e. GLR, and ANN). According to further explanation by (Yu, Wang and Lai, 2005), the new integrated approach is more accurate in comparison with the GLAR, and ANN individual systems. Similar to foreign exchange prediction, stock market forecasting is also a well-researched area. An HMM-based tool is developed by (Hassan and Nath, 2005). An HMM is used to scan for the similar patterns from the past data. Based on the previous trends, the forecasting is done by interpolating the adjacent log-likelihood values of the data sets. The results show the high potential that the HMM approach has in predicting stock exchange.

On the other hand, (Hassan, 2009) proposes a combined model of HMM and the Fuzzy models. The HMM identifies hidden data patterns, and fuzzy logic is used to generate a forecast value. The entire data space is partitioned based on the log-likelihood for each data string. They are used to generate the fuzzy rules. The performance of the proposed system is outperforming in comparison with ANN, and ARIMA. As a further improvement, (Cao *et al.*, 2015) proposes an addition to the methodology introduced by (Nguyen, 2016). In addition to the process historical data, they use the likelihoods of the model of the most

recent data set. Further, they use the developed models to predict stock closing prices of Apple, Google, and Facebook using single observation data and multiple observation data. It has concluded that the results from multiple observation data perform better in stock price predictions.

Also, the Maximum a Posteriori HMM approach is introduced by (Gupta and Dhingra, 2012) which is another prediction tool. Given historical data, this method can forecast stock values for the coming day. To train the continuous HMM, they utilize the high and low values in a given day and the fractional change in the stock value of the stock. A Maximum a Posteriori decision is made using the trained HMM. This approach has also shown the excellent potential of HMM in stock price prediction.

Real-time supervision of bioprocesses is very useful in quality control. A novel efficient modelling and supervision technique based on multiway partial least squares (MPLS) is presented by (Ündey, Tatara and Çınar, 2004). The method can predict the quality of the batch at the end of the growth. A real-time knowledge-based system (RTKBS) is employed for process monitoring, quality estimation, and fault diagnosis. Using a fed-batch penicillin production benchmark process simulator, they have validated the performance of the methodology.

A data-driven fault prediction method is proposed by (Wang *et al.*, 2018), which can quantify the degree of abnormality, based on probability density estimation. The method can be used to monitor the state of a complex system quantitatively. They first define an index to quantify the degree of abnormality. Next, a single slack factor multiple kernel SVM probability density estimation model is employed to improve the computational

efficiency. In addition to that, this improves the data mapping performance. The resulting model is capable of providing a rapid estimation with higher precision. The degree of abnormality is found to be accurately measurable by the abnormality index.

In most of the applications, the systems tend to show some characteristic signals before the actual appearance of a fault. Nevertheless, the extraction of those features for fault prediction process is challenging. A novel study is done by (Baek and Kim, 2018) to address this issue. They introduce two crucial definitions called symptom pattern and symptom period. Then they present a methodology for symptom pattern extraction. It collects all evidence from sensor signals related to faults can occur shortly. This study is based on the assumption that there is a period before the occurrence of a fault, which carries the characteristic symptoms to that particular fault. The proposed method is validated using a scenario related to abnormal cylinder temperature in a marine diesel engine and automotive gasoline engine knocking.

Process fault prediction and prognosis is currently a highly tricky area as it can reduce most of the risks including financial, health, and reputation. There are numerous studies have been done over the years to solve this problem in applications such as software, electronic circuits, cement manufacturing, distillation columns, computer disk drives, mechanical bearings, chemical and biological processes, metal smelting, semiconductor, stock market prediction and foreign exchange. Among the techniques used, ANN, GP, Bayesian Methods, Naïve Bayes sub-models, Naïve Bayes Classifier, HMM, GMM, PCA, KPCA, Decision Tree, k-NN, CAE, CVTA, LLNF, Particle Predictor, GLAR, Fuzzy Models, and MPLS can be found in most of the applications. It is clear that HMM-BN

combined fault prediction and prognosis is not explored and there is an excellent potential to outperform the available techniques. Therefore, the present work is aimed to propose a novel hybrid approach of fault prediction and prognosis. The method comprises two robust techniques: Hidden Markov and Bayesian Network. The remaining paper is organized as section two provide fundamentals of two techniques and how they are integrated. Section three details the methodology and its testing. Section 4 discusses results while section 5 presents the conclusions.

3.2 Preliminaries

3.2.1 Hidden Markov Model (HMM)

A Markov chain is a random process that involves several different states. There are relationships between the states due to the state transitions. Each of these transitions has an associated transition probability. Also, each state has an associated observation. The main characteristic of a Markov process is that the state transition to the next state only depends on the current state and not on any of the former states. The specialty of this technique is, the actual state sequence is not observable. Therefore, it is called the Hidden Markov model. HMM with a discrete output probability distribution can be represented as; $\lambda = \{A, B, \pi\}$; where λ is the model. On the other hand, $A = \{a_{ij}\}$, $B = \{b_{ij}(k)\}$, and $\pi = \pi_i$ stand for transition probability distribution; observation probability distribution; and initial state distribution respectively.

If S_i is a given state, parameters can be defined as; $a_{ij} = P(q_{t+1} = S_j | q_t = S_i)$, $1 \leq i, j \leq N$, $b_{ij}(k) = P(O_k | q_t = S_i)$, $1 \leq j \leq N$, $1 \leq k \leq M$, and $\pi_i = P(q_1 = S_i)$, $1 \leq i \leq N$. Here, q_t , N , O_k , and M stand for State at time t, Number of states, kth Observation, and

some sharp observations. Model λ can generate the probability of the observation sequence of visible states. The probability is calculated using equation [11] based on $b_{ij}(k)$.

$$P(O, \lambda) = \sum_{all\ S} \pi_{s_0} \prod_{T=0}^{T=1} a_{s_y} s_{t+1} b_{s_{t+1}}(O_{s_{t+1}}) \quad [11]$$

Some key algorithms are running in HMM, namely the k-means algorithm, the Expectation Maximization Algorithm (EM), and the Viterbi Algorithm.

k-means is almost a binary algorithm. This algorithm allows finding the cluster centers. This converges to local minima. It is required to know the number of cluster centers (*i. e. k* as an input). As the initiation, the value of k (i.e. number of clusters) is randomly guessed. As the repeating step, the data corresponds to the nearest cluster and then clusters are updated using corresponding data points. If the cluster is empty, the process re-starts at a random point until no change.

On the other hand, the EM algorithm is one who uses other probability distributions. EM is a probabilistic generalization which also allows finding the cluster centers. It modifies not only the shape of the clusters but also the co-variant matrix. This is probabilistically sound and can be proven that it converges in a log likelihood space. Similar to k-means, this gets converged to local minima. We need to know the number of cluster centers (*i. e. k*), similar to the k-means algorithm. $P(x) = \sum_{i=1}^k P(C = i) \cdot P(x|C = i)$ Where; $P(C = i) = \pi_i$ is the Prior probability to the cluster center, $P(x|C = i)$ is the Gaussian parameter for each of the individual Gaussian (i.e. μ_i, Σ_i where $i = 1, 2, 3 \dots$). The code can be divided into two sections namely Expectation Step (E-Step) and Maximization Step (M-Step).

In the E-Step, it is assumed that π_i, μ_i and Σ_i are known values. If e_{ij} is the Probability of j^{th} data point corresponds to cluster point i ;

$$e_{ij} = \pi. (2\pi)^{-M/2} |\Sigma|^{-1} \exp \frac{1}{2} (x_j - \mu_i)^{-1} \sum_i^{-1} (x_j - \mu_i) \quad [12]$$

Where; π stands for the prior probability to the cluster center, $(2\pi)^{-M/2} |\Sigma|^{-1}$ is the Normalizer and $\exp \frac{1}{2} (x_j - \mu_i)^{-1} \sum_i^{-1} (x_j - \mu_i)$ is the Gaussian expression.

In M-Step, π_i can be taken from $\sum_j e_{ij}/M$; μ_i can be taken from $\sum_j e_{ij} \cdot x_{Xj} / \sum_j e_{ij}$; and Σ_i can be taken from $\sum_j e_{ij} (x_j - \mu_i)^T (x_j - \mu_i) / \sum_j e_{ij}$. Here e_{ij} works as a soft correspondence of a data point which works as a weight for the calculation.

Defining the value of k is the next problem to be solved. This number is not known in real-world applications. However, it is assumed a constant. In practical cases, we guess the value of k and minimize the following expression, which is called the Log Likelihood.

$$LL = - \sum_j \log P(x_j | \sigma_1 \Sigma_1 k) + COST \times k \quad [13]$$

Here, LL is the Log-likelihood, and $COST$ is a constant penalty. Further, the posterior probability (i.e. $P(x_j | \sigma_1 \Sigma_1 k)$) is maximized of data. As shown in the equation, if the number of clusters of data is increased, the penalty will be high. Nevertheless, typically, this minimizes at a certain value of k .

Fault diagnosis and prediction system is proposed by (Li *et al.*, 2017), which is made up of three parts, namely: data preprocessing; degradation state detection, and fault diagnosis. For feature extraction, the wavelet transforms correlation filter is employed. To enhance the performance of HMM, they further propose an HMM-based semi-nonparametric

method by the probabilistic transition frequency profile matrix and the average probabilistic emission matrix. The proposed methodology is validated to be capable of identifying the system operating state and hence facilitate to predict the system behaviour.

3.2.2 Bayesian Networks (BNs)

BN is a type of probabilistic graphical models (GMs) (Ruggeri, Faltin and Kenett, 2007). These graphical structures can acquire the knowledge about the uncertain domain. Probabilistic causal relationships are demonstrated by the arrows, and each node connected by an arrow represents a random variable. To estimate the conditional dependencies in the graph, known statistical and computational methods are used. Hence, BNs combine principles from graph theory, probability theory, computer science, and statistics. Further, as per (Neapolitan, 2010), a BN model can be used to study the structures of gene regulatory networks. It can join in information from both prior knowledge and experimental data. BNs can be considered as a powerful tool for fault diagnosis. It has shown remarkable performance in FDD in work presented by (Amin, Imtiaz and Khan, 2018). Therefore, the predicted fault by HMM can be prognosed by BN.

As mentioned in (Wu *et al.*, 2017), The basic principles of BN are conditional independence and joint probability distribution:

$$P(V_1, V_2, \dots, V_k | v) = \prod_1^k P(V_i | v) \quad (i = 1, 2, \dots, k) \quad [14]$$

$$P(V_1, V_2, \dots, V_k | v) = \prod_1^k P(V_i / Parent(V_i)) \quad (i = 1, 2, \dots, k) \quad [15]$$

BNs based rare event prediction has been studied by (Cheon *et al.*, 2009) and (Cózar, Puerta and Gámez, 2017). BN has both causal and probabilistic semantics. Hence, it is ideal to combine actual process historical data, and background knowledge. To be used when there are not many details available about all possible values, (Cózar, Puerta and Gámez, 2017) have proposed a general-purpose decision support system tool. It consists of two BN models: one represents the failure-free behaviour of the system, and the other represents abnormal behaviours. This novel system is a robust tool that can be used for health management in industrial environments. The core of the system is a probabilistic expert system based on dynamic BNs. Fault detection is based on both conflict analysis and the likelihood-ratio test.

3.3 The Methodology

Under this topic, the pre-processing of data, training of HMM, prediction of abnormalities, training of BN, and prognosis of the predicted fault using BN are discussed. As illustrated in *Figure 3-3*, data is pre-processed to extract standard operating condition data, and the extracted data is used to establish HMM. The trained HMM is then used to generate conditional probabilities for BN. On the other hand, the PFD is used to develop the qualitative BN (i.e. the structure of BN), through the development of Signal Directed Graph (SDG). Then the CPTs of qualitative BN are filled using the generated conditional probabilities to establish the quantitative BN. The methodology is explained in detail in working paper (Pathmika and Khan, 2018a).

In using the established HMM and BN, the procedure illustrated in *Figure 3-4* is followed. The incoming real-time data is fed to the trained HMM, and the possible data string after

50 s is predicted. The predicted data strings are carefully assessed to detect potential future abnormalities. Once an abnormality is detected, that information is sent to the trained BN as likelihood evidence, and the cause for the particular abnormality is proposed.

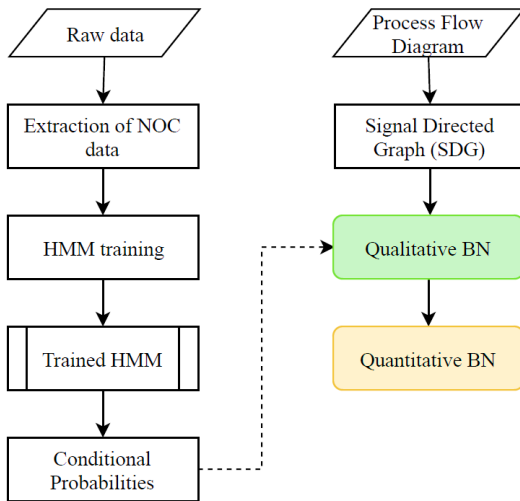


Figure 3-3: Establishing HMM and BN

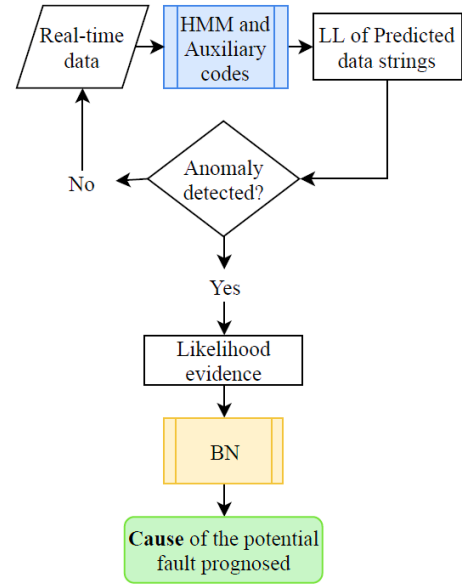


Figure 3-4: Overview of prediction and prognosis

3.3.1 Data preprocessing

Initially, there are ten datasets for ten different faults as shown in *Table 3-1*. For example, in dataset A, first 1000 data strings represent normal operating conditions. 1001 to 1500 data strings represent the respective faulty state. First, standard operating condition data are extracted from each data set and used to train the HMM. On the other hand, to generate a testing data set, $\pm 0.002\%$ of random noise for all 22 variables were added using a Matlab code. This noise level is gradually reduced to find the maximum possible noise level that the testing data set can go up to while keeping the accuracy of prognosis through BN.

3.3.2 Training of HMM and prediction of the n^{th} data string

The open source HMM toolbox for Matlab by (Murphy, 2005) is used in this study. *Figure 3-5* illustrates the training process of HMM. μ_0 and σ_0 are an initial assumption for mean for the mixture of Gaussians and Initial assumption for standard deviation for the mixture of Gaussians respectively. They are derived through *mixgauss_init* function, based on the inputs $Q, M, data, O, T, \text{ and } nex$.

Table 3-1: Faults in the TE process and their respective causes

Fault ID	Actual Root Cause
A	XMEAS (3)
B	XMEAS (9)
C	XMEAS (11)
D	XMEAS (9)
E	XMEAS (11)
F	XMEAS (4)
G	XMEAS (9)
H	XMEAS (11)
I	XMEAS (1)
J	XMEAS (19)

Then, $prior_0, transmat_0, mixmat_0,$ and $data$ along with $\mu_0,$ and σ_0 are updated by an EM algorithm to determine $LL, prior_1, transmat_1, \mu_1, \sigma_1,$ and $mixmat_1$. This provides a trained HMM which can give a higher log likelihood when it is provided with a data set with similar features. *Table 3-2* provides a description of each variable which is useful in training an HMM using HMM Toolbox developed by (Murphy, 2005). Further, *Figure 3-5* illustrates the connection between functions used in HMM tool box such as *mixgauss_init, mhmm_logprob, mixgauss_prob, and viterbi_path*. This figure is solely

presented to illustrate the information flow from one function to another. Detailed explanation on the function of HMM tool box is not presented for clarity.

In selecting the number of hidden states (Q), cross-validation technique is used. If there are N training samples and N parameters, a perfect score can be achieved, means free of errors. However, this can be done only for available data. That does not mean that the model is strong enough to predict the unseen data. For example, the stock price variation in the coming month can be given.

If the data are not correctly fitted, it is called under fit. Moreover, if all the noise is also taken into account, it is called overfitting. Therefore, this careful separation of correct data is essential. As a solution to this, the following approach can be taken.

The entire data set is separated into several portions, and one set is kept as a validation data set while the rest of the data are taken as training data. After that, the cost of each validation data set is considered. Through this method, we can choose the optimum number of states that give the highest validation accuracy.

A standard way of doing this is K-fold cross-validation. Here, the total data set is separated in to (say N) sets. Then any $(N - 1)$ sets can be selected as training data, and the remaining one can be used for validation. This N can be varied over a possible range, and the mean and standard deviation is used in making the decision. On top of this, the knowledge on the physical system can also be used in deciding the number of hidden states.

In training the HMM, the *test data* are fed as a 3D matrix to the HMM toolbox as shown in *Figure 3-6*. The number of observation sequences (T) is the number of rows in the data set while the number of different observations in a given data string (O) is the number of

columns in the data set. Also, there is a provision for datasets from similar processes called *nex*. In the current study, $nex = 1$.

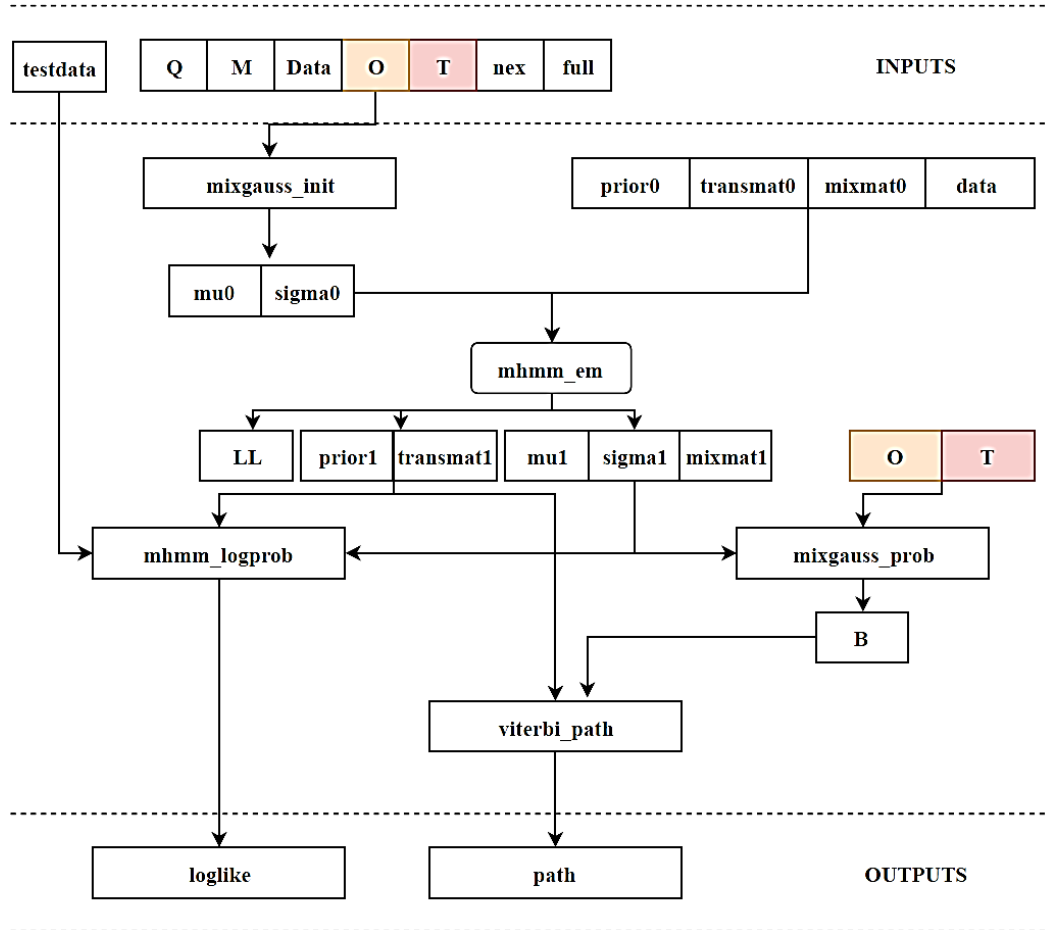


Figure 3-5: Training and application process of an HMM

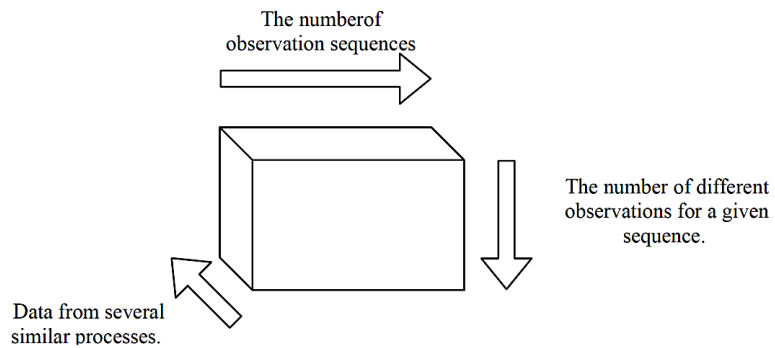


Figure 3-6: 3D matrix of data

Table 3-2: Description of notations used

Notation	Stands for	Description
Q	Number of hidden states	The optimum value of this to be determined by cross-validation
M	Number of mixtures of Gaussians	The optimum value of this to be determined by cross-validation
<i>Data</i>	Training data	This is the historical standard operating condition data of the process used to train the model
O	Size of a given vector-valued sequence	The number of different observations for a given sequence. i.e. the number of different sensors that give readings per unit time.
T	Length of a given vector-valued sequence	The number of observation sequences
Nex	Number of vector-valued sequences	The observations (i.e. T , and O) taken from similar processes
<i>Full/spherical/diag</i>	Co-variance matrix type	Trial and error selected optimum one
<i>mixgauss_init</i>	Function	Estimates initial parameters for a mixture of Gaussians using K-means algorithm.
μ_0	Initial parameter: mean	The initial assumption for mean for the mixture of Gaussians
μ_1	Updated mean	Updated through a <i>mixgauss_init</i> function for the mixture of Gaussians
σ_0	Initial parameter: standard deviation	The initial assumption for standard deviation for the mixture of Gaussians
σ_1	Updated standard deviation	Updated standard deviation through a <i>mixgauss_init</i> function for the mixture of Gaussians
$prior_0$	Initial state probability estimate	The initial assumption for state probability

<i>Prior1</i>	Updated state probability	Updated state probability through EM algorithm for the mixture of Gaussians
<i>transmat0</i>	Initial state transition probability	Assumed value for transition probability
<i>transmat1</i>	Updated state transition probability	Updated state transition probability through EM algorithm for the mixture of Gaussians
<i>mixmat0</i>	Initial Gaussian mixture matrix	Assumed values for the GM matrix
<i>test data</i>	New observations	Real-time incoming data can be fed through this variable
<i>LL</i>	Log-likelihood $\mathcal{L}(\Theta \kappa)$	$\mathcal{L}(\Theta \kappa) = \prod_{i=1}^N p(x_i \Theta) = p(\kappa \Theta)$
<i>mhmm_logprob</i>	Function	This evaluates the log likelihood of a trained model given test data. This computes the log likelihood of a data set using a (mixture of) Gaussians.
<i>mixgauss_prob</i>	Function	Evaluate the pdf of a conditional mixture of Gaussians
<i>viterbi_path</i>	Function	Determines the most likely hidden state path the system may take for the given set of observations.
<i>loglik</i>	Log likelihood of the test data	Determines the log likelihood of the given data sequences. i.e. likelihood of occurrence
<i>path</i>	Path of hidden states	Path of most probable hidden states for the given set of observations (i.e. test data)

Once the data is fed, the HMM can be trained. The training curve can be plotted as shown in *Figure 3-7*. It shows that the LL value gets consistent with the number of iterations. It means that the parameters estimated (*i. e.* λ) for HMM are consistently compatible with the training dataset. In brief, the HMM is trained for the given data set.

In addition to the above work done offline, the trained HMM is then used to predict the n^{th} data string by following the online procedure shown in *Figure 3-8*. Prediction with HMM requires experience on similar past incidents. Once the system understands the current state, it scans its memory (i.e. the LL value history) for a similar state, and the prediction is made concerning that. This is the central concept used in the prediction process.

If $t_{current}$ is the current time, up to $(t_{current} + n)^{th}$ data string is predicted. Once the actual time reaches $(t_{current} + n)$ the system reviews the predicted and observed data strings and creates knowledge which assists in predicting the next n data strings.

In the first step, three adjacent incoming data strings are fed to the HMM, and the respective LL values are evaluated. The LL values are then compared with the history of adjacent LL values. As a result, the most approximate three adjacent LL and the respective data string can be detected from past knowledge. Here it is assumed that the same pattern occurred in the history is likely to occur again. Based on that assumption, the n^{th} data string is predicted. This approach is a modified version of the work presented by (Hassan and Nath, 2005). As a further improvement of the prediction, the procedure illustrated in *Figure 3-9* is used. It can be observed from *Figure 3-10*, that the predicted and actual LL values follow a pattern which is almost similar. Here the solid line represents the actual variation of LL values during the previous data window and the dotted line represents the prediction made by the HMM for the same data window. Therefore, during the revision process shown in *Figure 3-9*, the correction of the mean value of predicted data concerning the previous data window is found to be satisfactory.

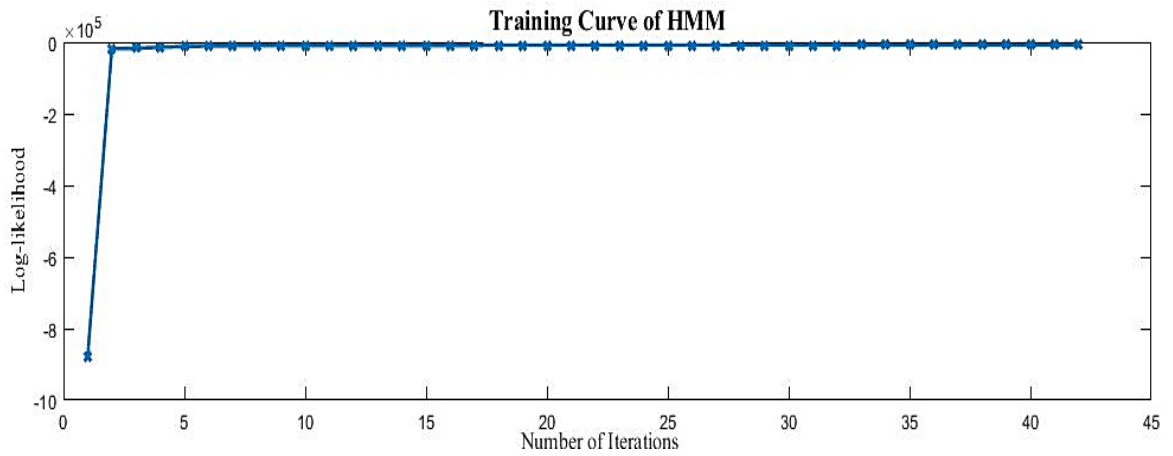


Figure 3-7: Training Curve for HMM

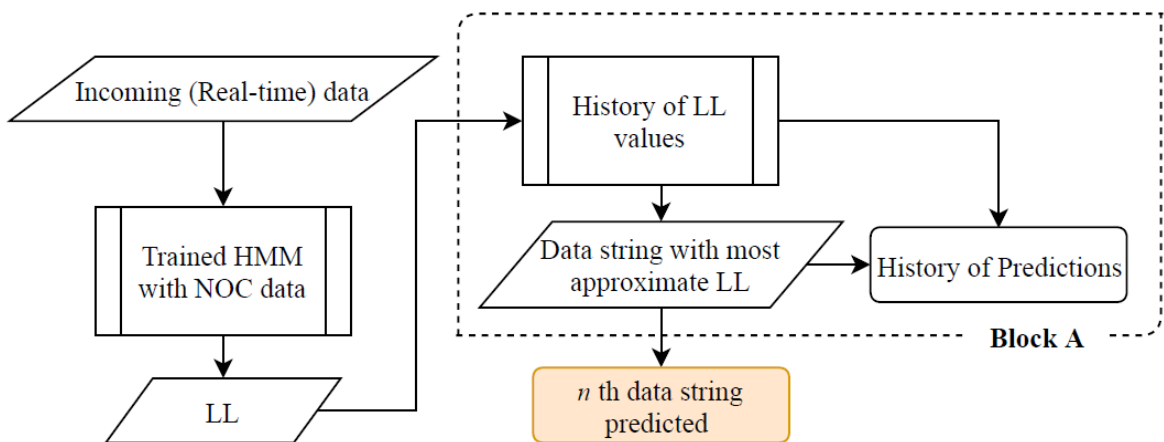


Figure 3-8: Prediction of n^{th} data string using trained HMM

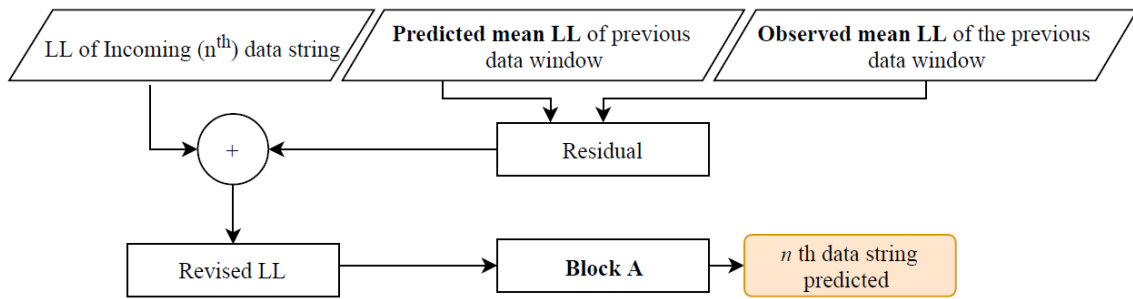


Figure 3-9: Refining process of the initial prediction

3.3.3 Development of structure and training of BN

This is the section that process knowledge is brought in to the prediction and prognosis process. The BN structure development begins in the Process Flow Diagram (PFD). The PFD can be converted to an SDG based on the knowledge of causality and effect. Then it is converted to a BN while maintaining the acyclic nature of BN.

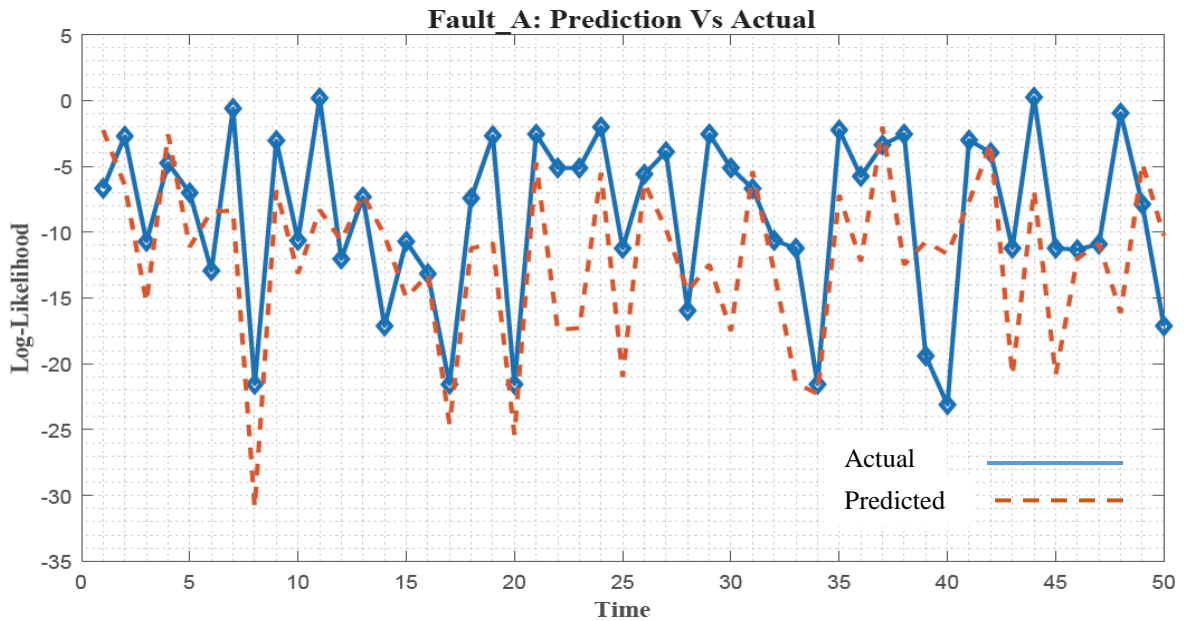


Figure 3-10: Predicted Vs. Actual LL of Fault A

A detailed explanation of the methodology is presented in the working paper (Pathmika and Khan, 2018a). On the other hand, the conditional probabilities derived through HMM are used to establish the CPTs of BN. The complete training process is also presented in (Pathmika and Khan, 2018a). The developed BN for the TE process is illustrated in *Figure 3-11*.

3.3.4 Introduction of Likelihood Evidence to the trained BN

As further mentioned in working paper (Pathmika and Khan, 2018a), in this step, the time of a fault is predicted using HMM. Assuming HMM detects the fault after a delay of t_0 number of seconds of actual introduction, data strings in the recent past (t_0 seconds) are also considered to calculate the probability of being all the nodes in 'Faulty State.' The value of t_0 depends on the complexity of the fault. If the LL value clearly shows a deviation soon after introducing the fault, t_0 can be a minimal value and vice versa. For example, $t_0 = 50$ s was taken.

3.3.5 Prognosis using BN

The methodology for prognosis is adopted from the study by (Amin, Imtiaz and Khan, 2018). Once the BN is updated with the likelihood evidence, the percentage change is evaluated in each node. If the highest percentage increase is in a root node, that node is taken as the cause for the particular problem. If not, the highest percentage change in the preceding successive parent node is taken as the cause.

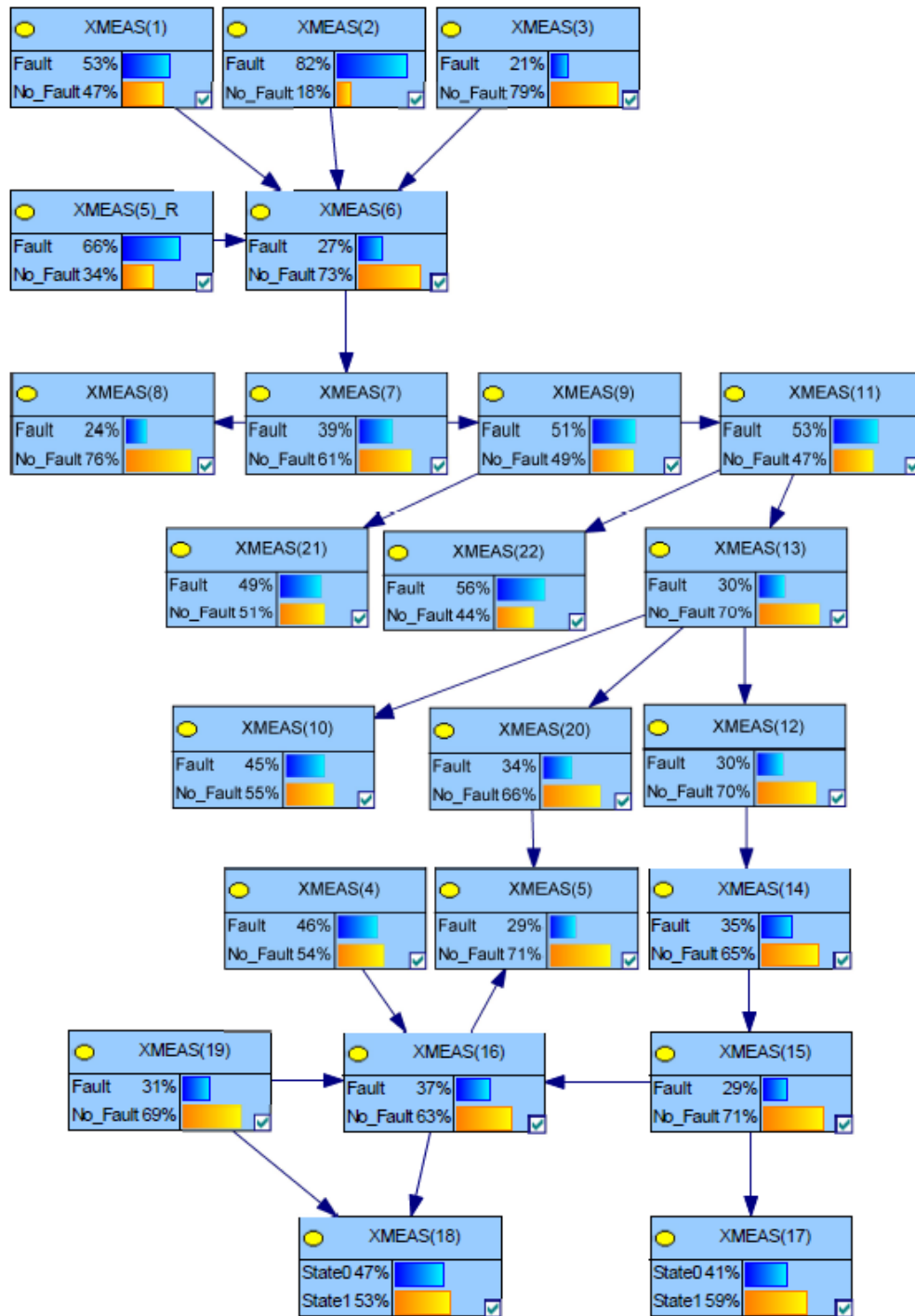


Figure 3-11: BN for the TE process (Pathmika and Khan, 2018a)

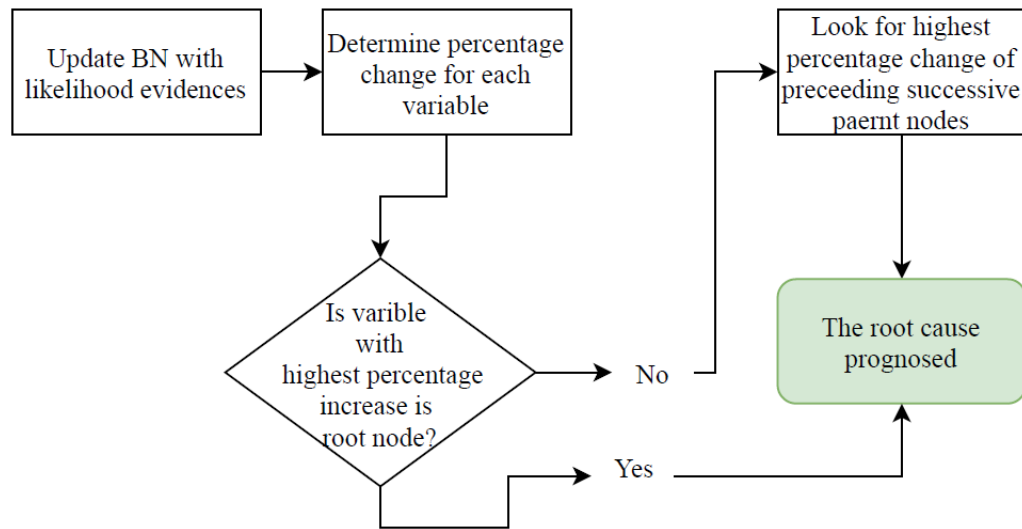


Figure 3-12: Procedure for prognosis of the fault

3.4 Results and Discussion

Under this topic, prediction and prognosis results of fault A are presented as a sample. Prediction on testing data set of fault A is illustrated in *Figure 3-13*. According to the same figure, fault A is detected at 995 s, which is introduced to the system at 1000 s. As mentioned in the working paper of (Pathmika and Khan, 2018a), the fault in the same data set grows to a detectable level at 1020 s. Therefore, this is a decent prediction of the abnormality. *Table 3-3* indicates the change in probabilities in each node after introducing the likelihood evidence. As illustrated, the highest percentage change is in XMEAS (3) which is the actual cause of fault A. This completes the prediction and prognosis of Fault A.

In a similar approach, the rest of the faults were also tested. It was observed that the prognosis performance varies for each fault. Therefore, the noise level was gradually reduced until the system accurately predicts the fault. *Table 3-4* shows the performance of

the prediction of each fault based on the maximum level of noise that can be handled by each system.

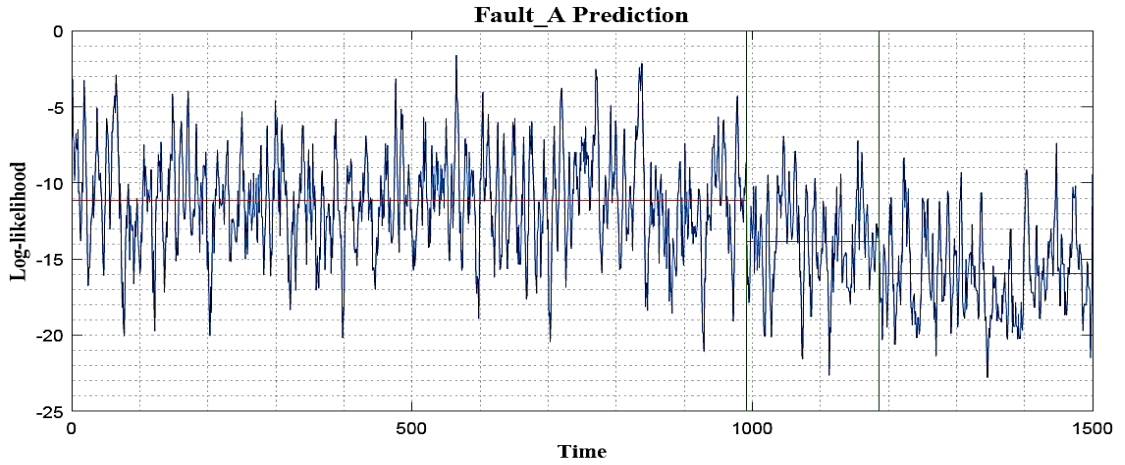


Figure 3-13: Fault A prediction using HMM

Table 3-3: Prognosis of fault A

Name of the Node	Initial Fault Probability	Final Fault Probability	Difference	Column 1
XMEAS_1	53	16	37	37
XMEAS_2	18	0	18	18
XMEAS_3	21	100	79	79
XMEAS_4	46	0	46	46
XMEAS_5_R	27	100	73	73
XMEAS_6	26	16	10	10
XMEAS_7	39	100	61	61
XMEAS_8	24	100	76	76
XMEAS_9	51	16	35	35
XMEAS_10	45	84	39	39
XMEAS_11	53	100	47	47
XMEAS_12	30	84	54	54
XMEAS_13	30	100	70	70
XMEAS_14	35	100	65	65
XMEAS_15	29	16	13	13
XMEAS_16	37	16	21	21
XMEAS_17	41	84	43	43
XMEAS_18	47	84	37	37
XMEAS_19	31	0	31	31
XMEAS_20	34	84	50	50
XMEAS_21	49	16	33	33
XMEAS_22	19	84	65	65

Table 3-4: Summary of results

Fault	Maximum Noise level can (for accurate prediction)	Actual Root Cause	Accurate Prediction
A	10^{-2}	XMEAS (3)	Yes
B	10^{-7}	XMEAS (9)	Yes
C	10^{-3}	XMEAS (11)	Yes
D	10^{-7}	XMEAS (9)	Yes
E	10^{-3}	XMEAS (11)	Yes
F	N/A	XMEAS (4)	No
G	10^{-8}	XMEAS (9)	Yes
H	10^{-6}	XMEAS (11)	Yes
I	10^{-8}	XMEAS (1)	Yes
J	N/A	XMEAS (19)	No

The data set used in the current study contains unexpected failures which are difficult to predict in comparison with gradually occurring faults. There is a little discussion about trend prognosis of abrupt faults available in the literature (Li *et al.*, 2014). The prediction using HMM mainly utilizes the experience of faults and the variation characteristics before a fault. In other words, it tracks the features (symptoms) that a system will show just before the start of a fault. Therefore, the proposed technique is recommended for systems which show some characteristics or symptoms before failure. It can be a pattern of variation of pressure or any physically measurable parameter. These are not rare in practical applications, and HMM is mighty in classifying those features.

The testing data used has $\pm 10^{-2}$ to $\pm 10^{-8}$ of variation in all 22 parameters in comparison with the training data. Some of the faults are not proposed adequately at higher noise levels. Faults A, C, and E were accurately prognosed even at comparatively higher noise levels. Because the system is composed of 22 sensors, the information of an abnormality may have

been diluted by the rest of the sensor readings and the failure detection of F, and J may have become unsuccessful.

3.5 Conclusions

A hybrid methodology of HMM-BN is proposed in process fault prediction and prognosis. HMM was used at the first stage to predict the fault using process historical data and real-time data. A BN is employed to determine a precise prognosis by reviewing the prediction made by HMM. The BN uses the process knowledge and inputs from HMM to perform the prognosis. Higher prognostic accuracy is achieved with this combined approach of HMM and BN. Depending on the complexity of the fault and the magnitudes of variations of each parameter, the accuracy of prediction varies. However, the proposed methodology can improve prognosis eight of the faults accurately with different levels of noise in testing data.

The present study contributes a new knowledge on process fault prediction and prognosis based on HMM and BN combined approach. The unique aspect of this study is the integration of HMM and BN for fault prediction and prognosis of process applications.

Faults F and J were predictable, as abnormalities, but the prognosis was not accurate. Prognosis of fault F and J, using the HMM-BN hybrid system will be a useful contribution.

The false alarm frequency needs to be reduced which is a practical problem that arises when implementing the proposed method in a real-world application. This can be done by implementing a suitable algorithm which defines when to indicate a potential fault. Further, it can be developed into a stand-alone computer programme based on Matlab.

References

- Amin, M. T., Imtiaz, S., & Khan, F. (2018). Process system fault detection and diagnosis using a hybrid technique. *Chemical Engineering Science*, *189*, 191–211. <https://doi.org/10.1016/j.ces.2018.05.045>
- Baek, S., & Kim, D. Y. (2018). Fault Prediction via Symptom Pattern Extraction Using the Discretized State Vectors of Multi-Sensor Signals. *IEEE Transactions on Industrial Informatics*, *3203(c)*. <https://doi.org/10.1109/TII.2018.2828856>
- Bekat, T., Erdogan, M., Inal, F., & Genc, A. (2012). Prediction of the bottom ash formed in a coal-fired power plant using artificial neural networks. *Energy*, *45(1)*, 882–887. <https://doi.org/10.1016/j.energy.2012.06.075>
- Cao, Y., Li, Y., Coleman, S., Belatreche, A., & McGinnity, T. M. (2015). Adaptive hidden Markov model with anomaly states for price manipulation detection. *IEEE Transactions on Neural Networks and Learning Systems*, *26(2)*, 318–330. <https://doi.org/10.1109/TNNLS.2014.2315042>
- Chelly, S. M., & Denis, C. (2001a). Applying Unsupervised Learning. *Medicine and Science in Sports and Exercise*, *33(2)*, 326–333. <https://doi.org/10.1111/j.2041-210X.2010.00056.x>
- Chelly, S. M., & Denis, C. (2001b). Applying Unsupervised Learning. *Medicine and Science in Sports and Exercise*, *33(2)*, 326–333. <https://doi.org/10.1111/j.2041-210X.2010.00056.x>
- Chen, M. Z., Zhou, D. H., & Liu, G. P. (2005). A new particle predictor for fault prediction of nonlinear time-varying systems. *Developments in Chemical Engineering and Mineral Processing*, *13(3–4)*.
- Cheng, F., Qu, L., & Qiao, W. (2017). Fault Prognosis and Remaining Useful Life Prediction of Wind Turbine Gearboxes Using Current Signal Analysis. *IEEE Transactions on Sustainable Energy*, *PP(99)*, 1. <https://doi.org/10.1109/TSTE.2017.2719626>
- Cheon, S.-P., Kim, S., Lee, S.-Y., & Lee, C.-B. (2009). Bayesian networks based rare event prediction with sensor data. *Knowledge-Based Systems*.

- <https://doi.org/10.1016/j.knosys.2009.02.004>
- Cózar, J., Puerta, J. M., & Gámez, J. A. (2017). An application of dynamic Bayesian networks to condition monitoring and fault prediction in a sensor system: A case study. *International Journal of Computational Intelligence Systems*, 10(1), 176–195.
- Deng, Z. (2018). *Proceedings of 2017 Chinese Intelligent Systems Conference* (Vol. 460). <https://doi.org/10.1007/978-981-10-6499-9>
- Di, Y., Jin, C., Bagheri, B., Shi, Z., Ardakani, H. D., Tang, Z., & Lee, J. (2018). Fault prediction of power electronics modules and systems under complex working conditions. *Computers in Industry*, 97, 1–9. <https://doi.org/10.1016/j.compind.2018.01.011>
- Ding, B., & Fang, H. (2017). Fault prediction for a nonlinear stochastic system with incipient faults based on particle filter and nonlinear regression. *ISA Transactions*, 68, 327–334. <https://doi.org/10.1016/j.isatra.2017.03.018>
- Fulufhelo V. Nelwamondo, T. M. and U. M. (2006). Early Classifications of Bearing Faults Using Hidden. *Information and Control*, 2(6), 1281–1299.
- Gao, Q., & Liu, W. (2017). Research and Application of the Distillation Column Process Fault Prediction based on the Improved KPCA, 247–251.
- Gupta, A., & Dhingra, B. (2012). Stock market prediction using Hidden Markov Models. *2012 Students Conference on Engineering and Systems, SCES 2012*. <https://doi.org/10.1109/SCES.2012.6199099>
- Hamerly, G., & Elkan, C. (2005a). Genetic Programming Approach for Fault Modeling of Electronic Hardware, 2(1), 1563–1569. <https://doi.org/doi:10.1109/CEC.2005.1554875>
- Hamerly, G., & Elkan, C. (2005b). Genetic Programming Approach for Fault Modeling of Electronic Hardware, 2(1), 1563–1569. <https://doi.org/doi:10.1109/CEC.2005.1554875>
- Hassan, M. R. (2009). A combination of hidden Markov model and fuzzy model for stock market forecasting. *Neurocomputing*, 72(16–18), 3439–3446. <https://doi.org/10.1016/j.neucom.2008.09.029>

- Hassan, M. R., & Nath, B. (2005). Stock market forecasting using a hidden Markov model: a new approach. *5th International Conference on Intelligent Systems Design and Applications (ISDA '05)*, 192–196. <https://doi.org/10.1109/ISDA.2005.85>
- Hu, Y., Deng, X., Cao, Y., & Tian, X. (2017). Dynamic process fault prediction using canonical variable trend analysis. *2017 Chinese Automation Congress (CAC)*, 2015–2020. <https://doi.org/10.1109/CAC.2017.8243102>
- Hughes, G. F., Murray, J. F., Kreutz-Delgado, K., & Elkan, C. (2002). Improved disk-drive failure warnings. *IEEE Transactions on Reliability*, *51*(3), 350–357. <https://doi.org/10.1109/TR.2002.802886>
- Illias, H. A., Chai, X. R., Bakar, A. H. A., & Mokhlis, H. (2015). Transformer early fault prediction using combined artificial neural network and various particle swarm optimization techniques. *PLoS ONE*, *10*(6), 1–16. <https://doi.org/10.1371/journal.pone.0129363>
- Jahani, K., & Razavi, J. (2016). Application of Computer Aided Engineering Tools in Performance Prediction and Fault Detection of Mechanical Equipment of Mining Process Line, *10*(7), 1295–1299.
- Jiang, B.-L., Wey, C.-L., & Fan, L.-J. (1988). FAULT PREDICTION FOR ANALOG CIRCUITS *, *7*(1).
- Kim, J., Han, Y., & Lee, J. (2016). Euclidean Distance Based Feature Selection for Fault Detection Prediction Model in Semiconductor Manufacturing Process, *133*, 85–89.
- Kim, M. J., Jiang, R., Makis, V., & Lee, C. G. (2011). Optimal Bayesian fault prediction scheme for a partially observable system subject to random failure. *European Journal of Operational Research*, *214*(2), 331–339. <https://doi.org/10.1016/j.ejor.2011.04.023>
- Li, C., Wei, F., Wang, C., & Zhou, S. (2017). Fault diagnosis and prediction of a complex system based on Hidden Markov model. *Journal of Intelligent & Fuzzy Systems*, *33*(5), 2937–2944. <https://doi.org/10.3233/JIFS-169344>
- Li, L., Wang, Z., Liu, Z., & Bu, S. (2014). Trend prognosis of aero-engine abrupt failure based on affinity propagation clustering. In W. Jinsong (Ed.), *In Proceedings of the*

- First Symposium on Aviation Maintenance and Management* (Vol. II, pp. 13–22). Berlin, Heidelberg.: Springer.
- Ma, J., & Xu, J. (2015). Fault Prediction Algorithm for Multiple Mode Process Based on Reconstruction Technique, 2015. <https://doi.org/10.1155/2015/348729>
- Murphy, K. (2005). Hidden Markov Model (HMM) Toolbox for Matlab. Retrieved July 19, 2018, from <https://www.cs.ubc.ca/~murphyk/Software/HMM/hmm.html>
- Murray, J. (2003). Hard drive failure prediction using non-parametric statistical methods. *Proceedings of ICANN/ ...*, (1). Retrieved from <http://hebb.mit.edu/people/jfmurray/publications/Murray2003.pdf>
- Neapolitan, R. E. (2010). Learning Bayesian Networks. *International Journal of Data Mining and Bioinformatics*, 4, 505–519. <https://doi.org/10.1016/j.jbi.2010.03.005>
- Nguyen, N. (2016). Stock Price Prediction using Hidden Markov Model, 1–20. Retrieved from https://editorialexpress.com/cgi-bin/conference/download.cgi?db_name=SILC2016&paper_id=38
- Pathmika, M., & Khan, F. (2018). *Dynamic process fault detection and diagnosis based on a combined approach of hidden Markov and Bayesian network model*. St. John's.
- Predix. (2016a). Machine Learning. Retrieved from <https://www.predix.io/analytics>
- Predix. (2016b). Machine Learning.
- Ramana, E. V., Sapthagiri, S., & Srinivas, P. (2017). Data Mining Approach for Quality Prediction and Fault Diagnosis of Injection Molding Process. *Indian Journal of Science and Technology*, 10(17), 1–7. <https://doi.org/10.17485/ijst/2017/v10i17/112580>
- Ruggeri, F., Faltin, F., & Kenett, R. (2007). Bayesian Networks. *Encyclopedia of Statistics in Quality & Reliability*, 1(1), 4. <https://doi.org/10.1002/wics.48>
- Sadeghian, M., & Fatehi, A. (2011). Identification, prediction and detection of the process fault in a cement rotary kiln by locally linear neuro-fuzzy technique. *Journal of Process Control*, 21(2), 302–308. <https://doi.org/10.1016/j.jprocont.2010.10.009>
- The Mathworks. (2016). *Introducing Machine Learning What is Machine Learning. Perspectives on Ontology Learning*.

- Ündey, C., Tatara, E., & Çinar, A. (2004). Intelligent real-time performance monitoring and quality prediction for batch/fed-batch cultivations. *Journal of Biotechnology*, *108*(1), 61–77. <https://doi.org/10.1016/j.jbiotec.2003.10.004>
- Wang, H. Q., Cai, Y. N., Fu, G. Y., Wu, M., & Wei, Z. H. (2018). Data-driven fault prediction and anomaly measurement for complex systems using support vector probability density estimation. *Engineering Applications of Artificial Intelligence*, *67*(December 2016), 1–13. <https://doi.org/10.1016/j.engappai.2017.09.008>
- Wu, J., Zhou, R., Xu, S., & Wu, Z. (2017). Probabilistic analysis of natural gas pipeline network accident based on the Bayesian network. *Journal of Loss Prevention in the Process Industries*, *46*, 126–136. <https://doi.org/10.1016/j.jlp.2017.01.025>
- Yu, L., Wang, S., & Lai, K. K. (2005). A novel nonlinear ensemble forecasting model is incorporating GLAR and ANN for foreign exchange rates. *Computers and Operations Research*, *32*(10), 2523–2541. <https://doi.org/10.1016/j.cor.2004.06.024>
- Zhang, S., Wang, Y., Liu, M., & Bao, Z. (2017). Data-Based Line Trip Fault Prediction in Power Systems Using LSTM Networks and SVM. *IEEE Access*, *6*, 7675–7686. <https://doi.org/10.1109/ACCESS.2017.2785763>
- Zhao, Y., Li, D., Dong, A., Kang, D., Lv, Q., & Shang, L. (2017). Fault prediction and diagnosis of wind turbine generators using SCADA data. *Energies*, *10*(8), 1–17. <https://doi.org/10.3390/en10081210>

CHAPTER 4 : AUXILIARY CODES FOR FAULT PROGNOSIS PROCESS OF TENNESSEE EASTMAN PROCESS USING A HYBRID MODEL (CPL1.0)³

Co-authorship Statement

Galagedarage Don Mihiran Pathmika is the principal author of this thesis. He has undertaken the research and prepared the first draft. Professor Faisal Khan, the co-author of the manuscript, shared the problem and conceptualized the methodology. In addition, Prof. Khan contributed by reviewing, and revising the manuscript. He also guided the author throughout the entire process of the methodology development, testing, validation and its application development. The software code and analysis of results were solely contributed by the principal author, and the results were validated for correctness by the co-author.

Abstract

CPL1.0 is a Matlab code which can generate fault predictions of Tennessee Eastman (TE) process. It facilitates the calculation of Conditional Probabilities (CP), Prior Probabilities (PP), and Likelihood Evidences (LE) which are useful in establishing the Bayesian Network (BN), which is later used in fault prognosis. Determination of the CP, PP, and LE is the most time-consuming component in the BN establishing process. Hence, users of this code can follow the methodology proposed for effective use of their time for new contributions. Sensor readings and their respective log-likelihood values are used to

³ This chapter is submitted as a manuscript to the journal of SoftwareX and currently under review

determine CP, PP, and LE. The code is presented in such a way that it can be adopted to any BN structure with minimum alterations. Further, HMM can be easily replaced with any other machine learning technique. Therefore, CPL1.0 is a facilitator to communicate between BN and the other machine learning technique in a hybrid fault prediction and prognosis system.

Keywords: *Conditional Probabilities; Prior Probabilities; Likelihood Evidence; Bayesian Networks*

4.1 Motivation and Significance

Use of hybrid systems has shown enhanced performance in fault prediction and prognosis. Bayesian Networks (BN) combined with other machine learning techniques is a current trend and many combinations still to be explored. Establishing the BN, based on the outputs of fellow machine learning technique, is a bit of a challenge as extensive coding is involved. This is mainly because of the conditional probability tables to be established in BNs. A systematic approach can simplify the problem of coding which can save a lot of valuable time that a scholar can effectively utilize to make their original contributions.

The importance of the current code is that it can generate the required information to establish BN, based on the outputs of the Hidden Markov Model (HMM). The code is specifically written for prediction and prognosis of potential faults in the Tennessee Eastman (TE) process based on a hybrid approach of HMM and BN. Nevertheless, it can be easily altered to fit any type of BN by using the basic functions available in Matlab software. Therefore, the code will contribute to the process of scientific discovery in the future by solving an intermediate problem of bridging the BN with other machine learning techniques, which will be a recurrent step in future work.

4.2 Software Description

CPL1.0 is a tool to determine essential inputs to a BN, which is 100% Matlab based code. It helps to combine BNs with another machine learning technique. The whole code is available on GitHub repository thus possible to clone or fork.

4.3 Software Architecture

To use the proposed code, the user should have access to Matlab software and HMM toolbox developed by Kevin Murphy. Once the HMM toolbox is loaded in Matlab, following the method mentioned in (Murphy, 2005), the CPL1.0 code can be used according to the procedure shown in . CPL1.0 is responsible for the operations mentioned in the shaded boxes in *Figure 4-1*. The user can feed the process normal operating condition (NOC) data to train the HMM. The trained HMM and a different set of NOC data to be used to generate a data history along with the respective log-likelihood (LL) values. The data history and the LL values are then used to generate the Conditional Probability Tables (CPT), and the respective data strings (i.e. data history) are used to generate the prior probabilities.

On the other hand, the real-time process data and the trained HMM is used for fault prediction and the generation of likelihood evidences. Later, all three types of information need to be manually fed to the BN, which has a structure developed using GeNIe. The user can just copy and paste the respective data strings into the CPTs of BN. Completion of this step facilitates making decisions on the root cause of the potential fault. The proposed code is used in the working paper (Pathmika and Khan, 2018b). The transfer of data from Matlab to GeNIe can be eliminated by using the BayesNet toolbox by (Kevin Murphy, 2001).

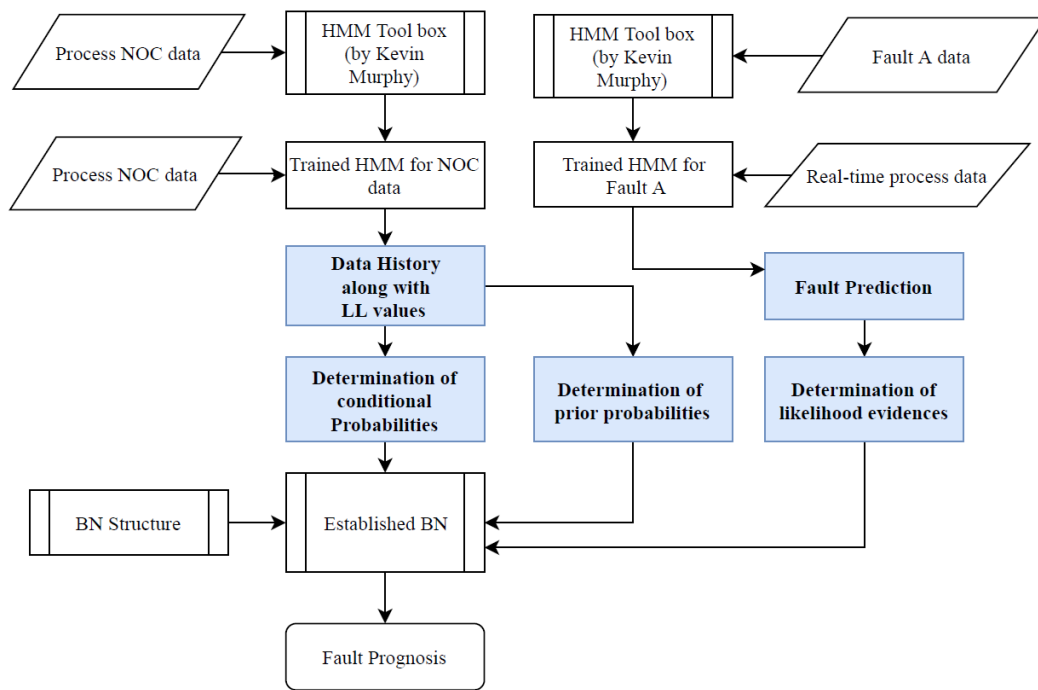


Figure 4-1: The procedure to use CPL1.0

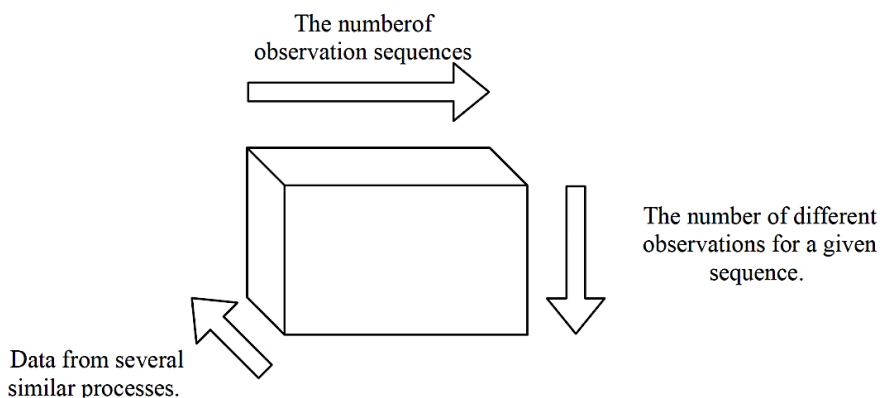


Figure 4-2: The input data format (Chapter 3, Figure 3-6)

4.4 Sample code snippets analysis.

A history of loglikelihood values (i.e. *loglikeALL*) is developed using the *mhmm_logprob* function in HMM toolbox. Next, the lower and higher limits of the safe zone are defined

using the code presented in equations [16] and [17] which are denoted by aNL , and aNH respectively. Here N stands for the number of the node. Here PH denotes the history of NOC data and the respective LL while r denotes a constant that can be altered to get a suitable fluctuation of states. If a parameter is in the Danger Zone, it will be considered as a potential fault hence given the designation F . If not, the designation N is given.

$$a1L = \text{mean}(PH(:, 1)) + (\text{min}(PH(:, 1)) - \text{mean}(PH(:, 1)))/r \quad [16]$$

$$a1H = \text{mean}(PH(:, 1)) + (\text{max}(PH(:, 1)) - \text{mean}(PH(:, 1)))/r \quad [17]$$

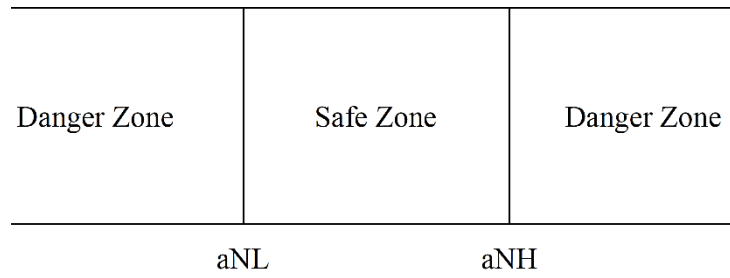


Figure 4-3: Safe and danger zones of each parameter reading (Chapter 2, Figure 2-10)

In a BN, for a given node with six parent nodes, there will be 2^6 number of conditional probability values to be determined. In other words, the combinations will vary from $F F F F F F$ to $N N N N N N$. The probabilities for each possible combination are determined using the code. Further, the prior probabilities are calculated for root nodes considering the history. Most importantly, when calculating the likelihood evidences, the CPL1.0 considers 50 data strings backwards from the point where the abnormality is detected. It is done in order to determine a most probable state combination at the point of detection.

In predicting, the code scans the history of LL values which shows three adjacent closely similar cases to the most recent three LL values. Then the next data string is predicted assuming a similar pattern to the LL history. Noise is added to each parameter of the original data set in the range of $10^{-2}\%$ to $10^{-8}\%$ of their respective values. Then the system is tested for the prediction performance for different noise levels.

4.5 Illustrative Example

Figure 4-4 illustrates the training curve of HMM using NOC data. The LL value approximately becomes a constant after 10 iterations. During this process, the parameters of HMM get tuned up such that it gives the best explanation to the training dataset.

Initially, the conditional probabilities and prior probabilities are determined using the code and then used to establish the BN. This component is a calculation done based on the data fed into CPL1.0. The user is recommended to run the code with the use of Matlab software and HMM Toolbox by Kevin Murphy. The output is a somewhat large matrix of data which contains the conditional and prior probability values. The user can just copy and paste the values in the correct cell of GeNIe. This is not a difficult task as the values are in the same pattern of GeNIe CPT table cells.

Once a new data string is provided to the trained HMM, it calculates the LL accordingly and plots the curve similar to the one shown in *Figure 4-5*. It illustrates the point where the system identifies the abnormality (i.e. 1130 s). On the other hand, *Figure 4-6* illustrates the prediction of abnormality which occurs around 995 s. This is a decent prediction of the fault for a testing dataset with added noise of $10^{-2}\%$.

The predicted fault is then used to determine the likelihood evidences, which are later fed to BN. This leads to determine the percentage changes in failure probability in each node. The detailed procedure is explained in the working paper by (Pathmika and Khan, 2018b)



Figure 4-4: The training curve of HMM (Chapter 3, Figure 3-7)

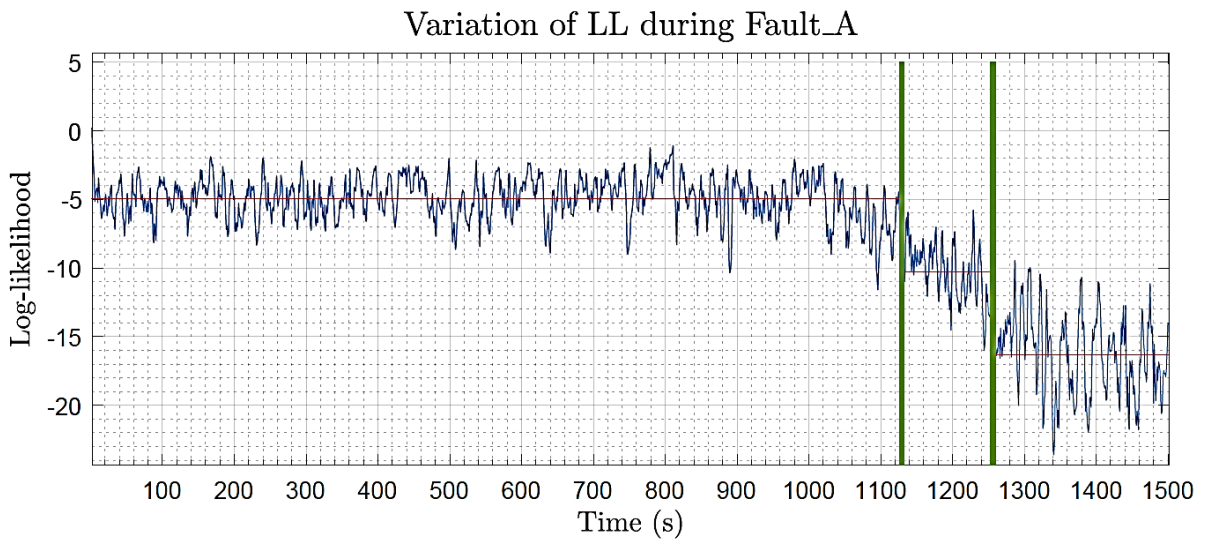


Figure 4-5: The actual LL variation of fault

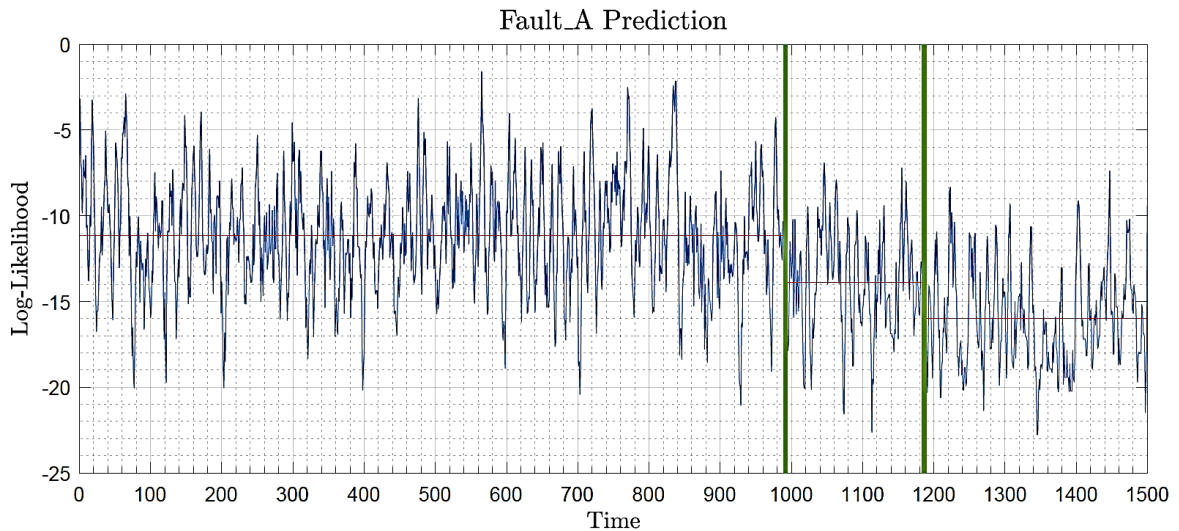


Figure 4-6: Predicted LL variation of the fault with 10^{-2} % of noise

4.6 Impact

With the availability of CPL1.0, new research problems on fault prediction and prognosis using BNs can be pursued conveniently. There are many different combinations still to be explored in this area. The proposed code will simplify a major recurrent step in that process. The improvement made through CPL1.0 code can be considered as a solution given to a common problem faced by the researchers who work related to BN. Future researchers can develop CPL1.0 up to a Matlab based stand-alone software. It can be made flexible to train any given BN, based on the outputs of the fellow machine learning technique. On the other hand, those who solely work on predictions using HMM will also get the benefit as the initial part of CPL1.0 deals with fault prediction using HMM.

The current practice of establishing CPTs include subjective approaches such as expert judgement. Nevertheless, the proposed code provides a better objective and transparent approach in determining the conditional probability values. Hence, rather than allocating

time on a problem that one has already solved, people can reuse this code while contributing to its development.

In addition to the intended user group of this code, the researchers who work in the areas of finance, weather forecasting, maintenance, and medical fields will also get the benefit. Therefore, the proposed code will have a wide spread of users. Further, this study provides an excellent example of how to apply HMM in a given problem. It will be valuable for HMM users as there is a very limited number of documented applications available in the literature.

Table 4-1: Code Meta Data

Nr	Code metadata description	
C1	Current code version	CPL1.0
C2	Permanent link to code/repository used of this code version	https://github.com/mihranpathmika/CP L1.0
C3	Legal Code License	MIT License
C4	Code versioning system used	none
C5	Software code languages, tools, and services used	Matlab R2018a
C6	Compilation requirements, operating environments & dependencies	Matlab R2015 onwards
C7	If available Link to developer documentation/manual	https://github.com/mihranpathmika/CP L1.0
C8	Support email for questions	Official: gdmpathmika@mun.ca Personal: mihranpathmika@gmail.com

4.7 Conclusions

CPL1.0 is a Matlab based software code which can be used to bridge BNs with other machine learning techniques. It provides a transparent and objective technique in establishing the CPTs of a BN which is somewhat challenging for BN users. On the other hand, CPL1.0 facilitates fault prediction and prognosis, with the use of HMM and BayesNet toolboxes introduced by Kevin Murphy. It has a wide spread of applications in several different fields hence a large number of potential users.

4.8 References

- Kevin Murphy. (2001). How to use the Bayes Net Toolbox. Retrieved September 3, 2018, from <http://www.cs.ubc.ca/~murphyk/Bayes/usage.11dec01.html>
- Murphy, K. (2005). Hidden Markov Model (HMM) Toolbox for Matlab. Retrieved July 19, 2018, from <https://www.cs.ubc.ca/~murphyk/Software/HMM/hmm.html>
- Pathmika, M., & Khan, F. (2018). *Process fault prediction and prognosis using a hybrid model*. St. John's.

CHAPTER 5 : SUMMARY AND CONCLUSION

Early detection or prediction of a fault, along with the respective root cause identification is extremely valuable in terms of process operations and safety concerns. A validated solution methodology for the problem is proposed through this thesis based on a hybrid approach of fully data-driven and fully knowledge driven techniques namely HMM and BN respectively.

As the first step, a method was proposed to detect the fault using HMM and diagnose the root cause using BN. It showed successful early detection for 100% of the faults and successful diagnosis for 80% of the faults.

As the second step, the methodology was further developed to predict faults even with some degree of added noise to the original data sets. It also predicted the faults with a 100% success and diagnosed the fault with an 80% of accuracy.

While providing a transparent methodology for the entire process, this study introduced a software code (CPL1.0), which can be effectively used in future researches related to hybrid FDD or FPP systems which includes BNs. The introduced software code and the procedure will save a considerable amount of time that a researcher allocates to establish CPTs of BN, with the use of outputs of other machine learning techniques.

In conclusion, the thesis proposes a successful solution methodology for FDD and FPP of process engineering applications in safety perspective. It detects and predicts 100% of the abnormalities while accurately diagnose and prognose 80% of the selected faults of benchmark TE process. Further, it presents a valuable software tool segment which has a potential to be used in future related research.

APPENDIX A: DETECTION OF FAULTS FROM FAULT A TO J

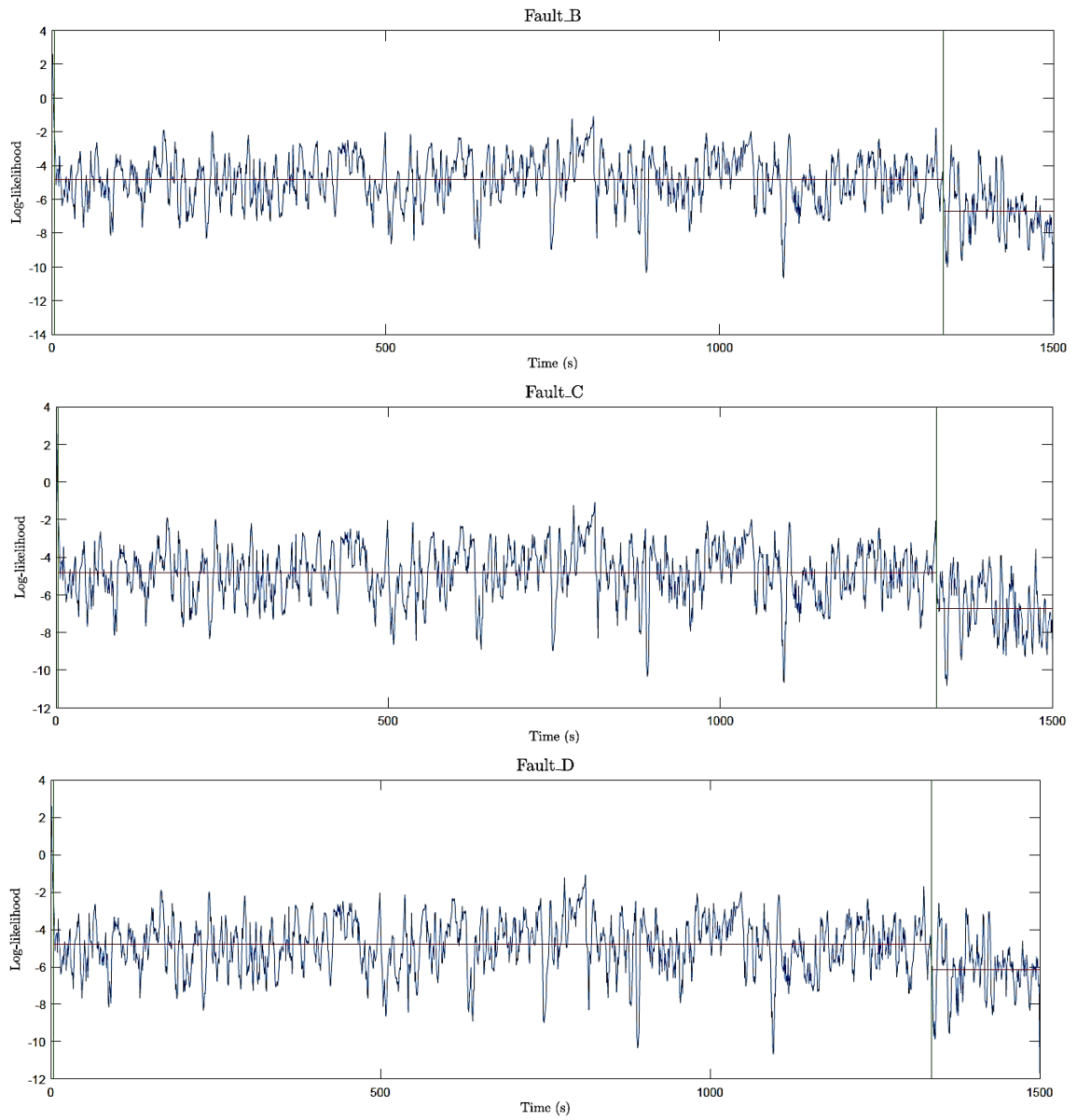


Figure A-1: Detection of fault B, C, and D

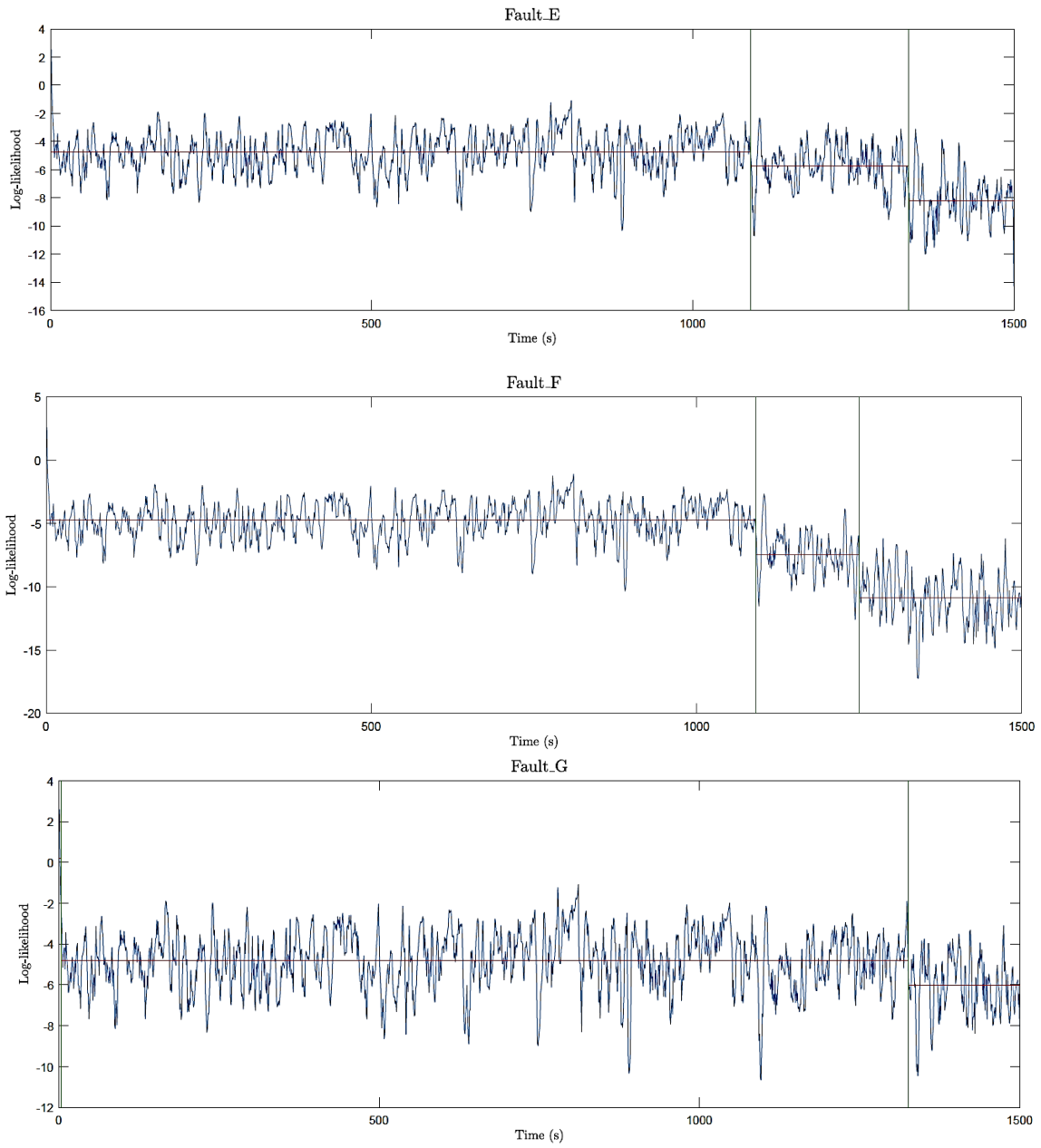


Figure A-2: Detection of fault E, F, and G

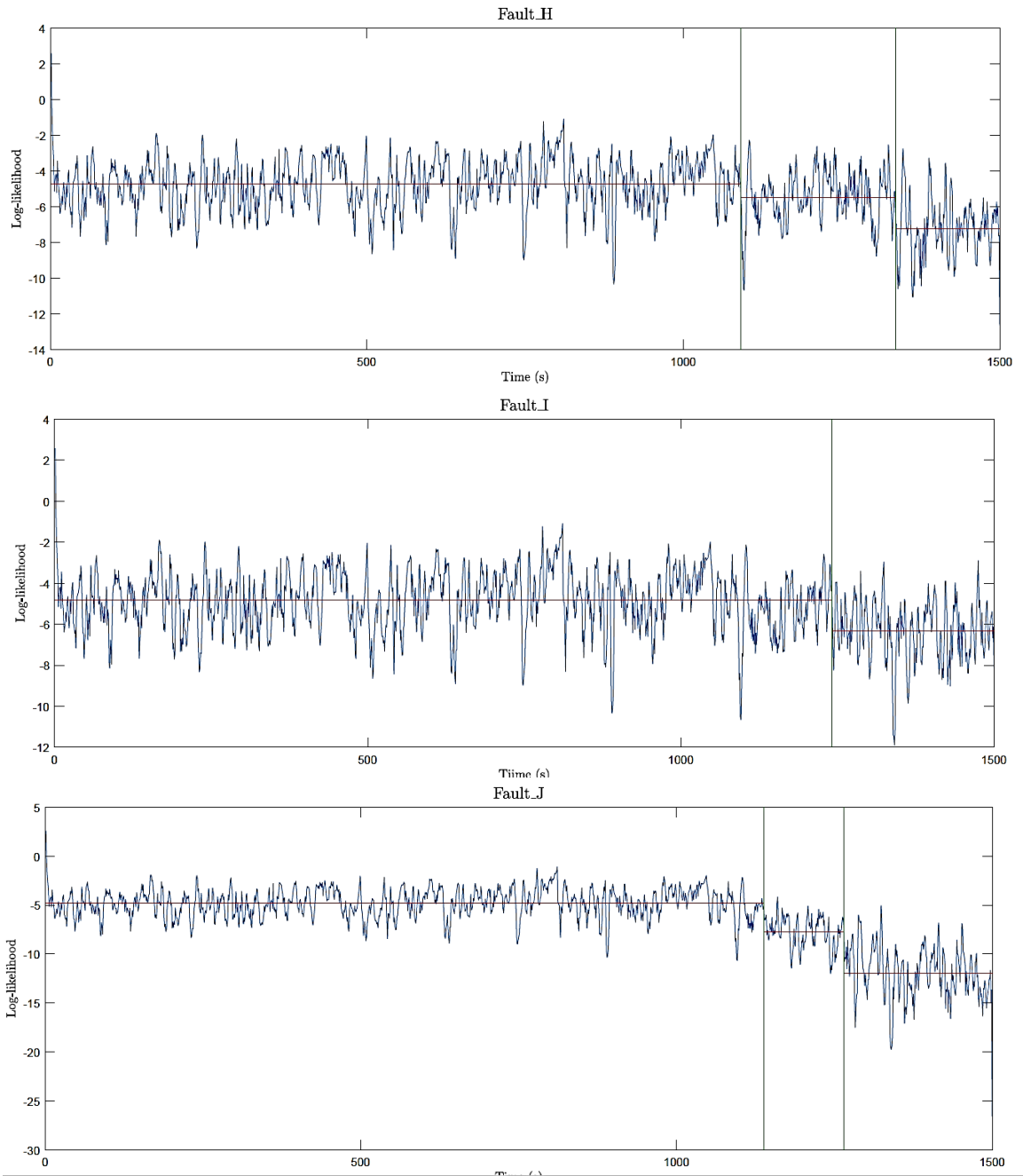


Figure A-3: Detection of fault H, I, and J

APPENDIX B: DETECTION OF FAULTS FROM FAULT A TO J

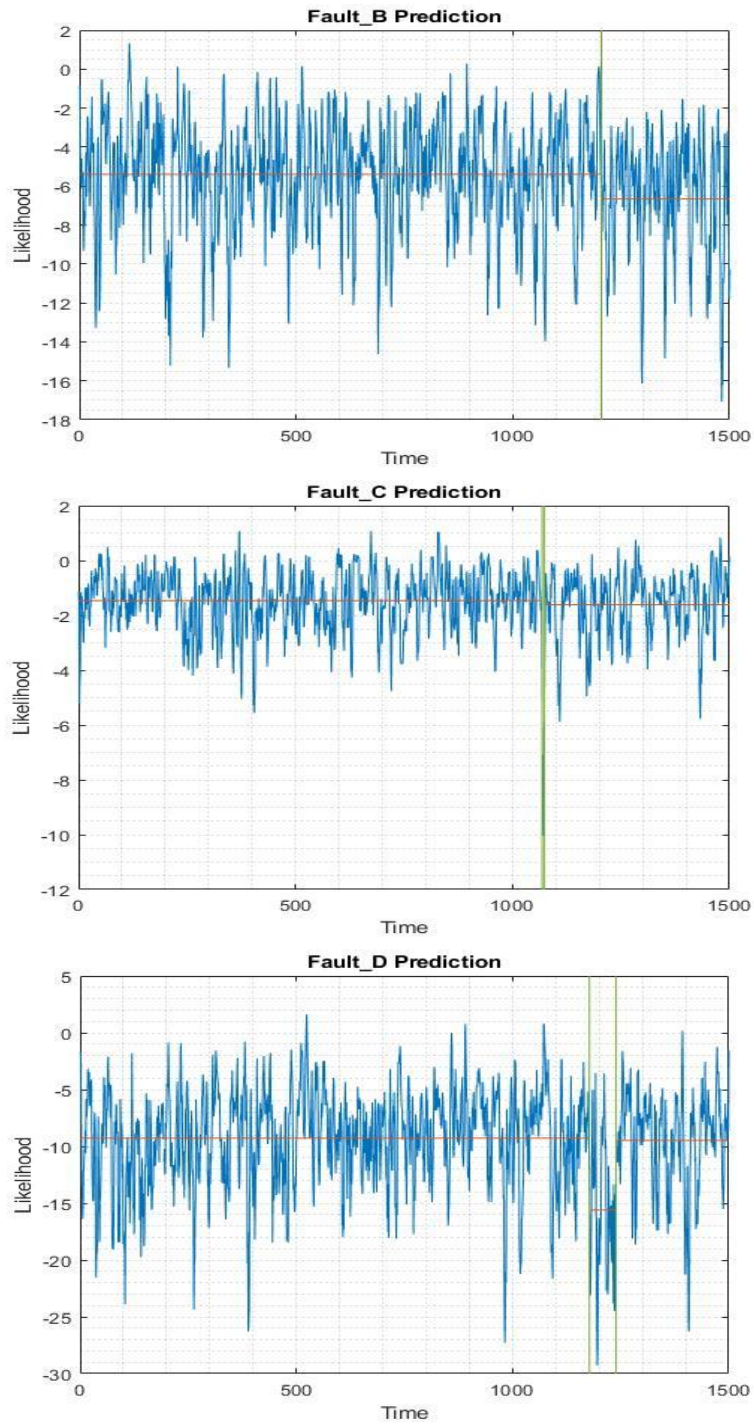


Figure B-1: Prediction of fault B, C, and D

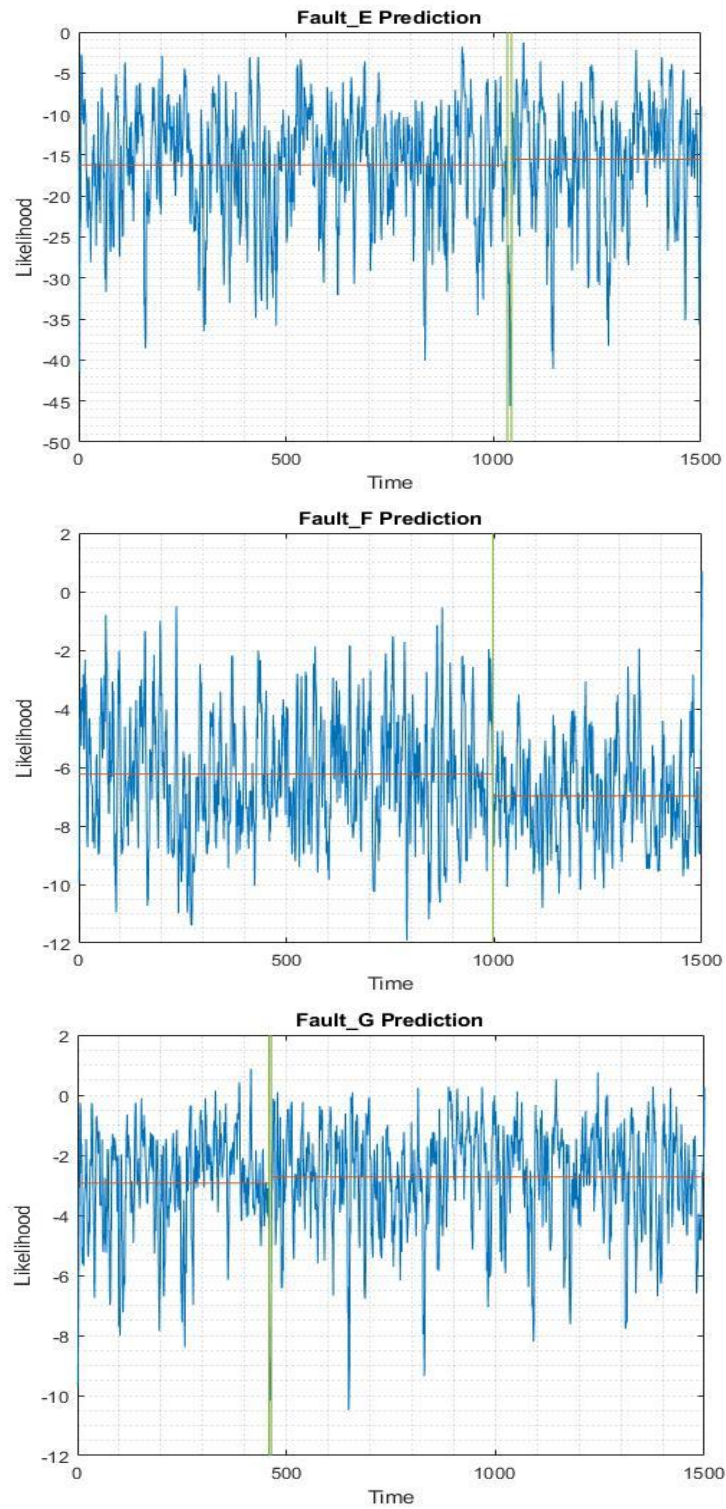


Figure B-2: Prediction of fault E, F, and G

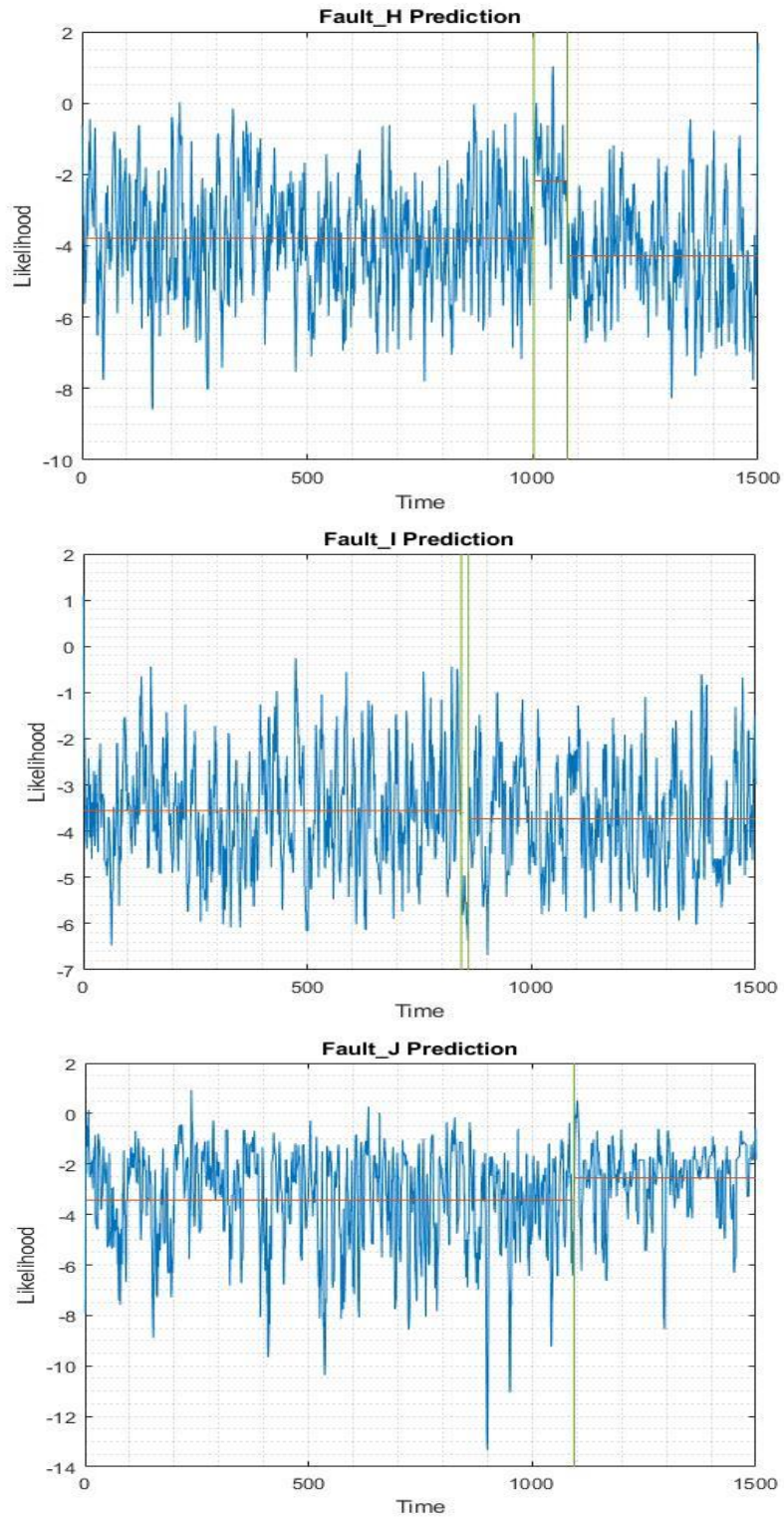


Figure B-3: Prediction of fault H, I, and J