Final Scientific/Technical Report or Progress Report

Project Title:	VOLTTRON COMPATIBLE WHOLE BUILDING ROOT-FAULT DETECTION AND DIAGNOSIS
Award Number:	DE-EE0007135
Approved Project Period:	Oct. 1st 2015 to September 30th 2018
Recipient:	Drexel University
DOE Project Team:	DOE Contracting Officer: Geoffrey I. Walker
DOE Project Officer:	Technology Manager: Marina Sofos
	Technology Project Officer: Mohammed J. F. Khan
	Project Engineer: Michael Geocaris; Michael Wofsey
Principal Investigator (PI):	Jin Wen
	Professor
	Email: jinwen@drexel.edu
	Phone: 215-895-4911
Working Partners:	KGS Buildings; Dr. Zheng O'Neill; Dr. Teresa T. Wu
Submitted by:	Dr. Jin Wen and Yimin Chen
	Drexel University, Philadelphia, PA
	December 2018

ACKNOWLEDGMENTS

This material is based upon work supported by the U.S. Department of Energy's Office of Energy Efficiency and Renewable Energy (EERE) under the Building Technologies Office under Award Number DE-EE0007135. Assistance from EERE program officers Mohammed J. F. Khan, Marina Sofos, Michael Geocaris and Michael Wofsey is greatly appreciated.

Special thanks are given to Drexel University's facility department, especially Mr. William Taylor, for their kind help during this project.

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

Executive Summary

Buildings consume more than 40% of primary energy in the U.S. Malfunctioning sensors, components, and control systems, as well as degrading components in Heating, Ventilating and Air-conditioning (HVAC) and lighting systems are main reasons for energy waste and unsatisfactory indoor environment. Field studies have shown that an energy saving of 5-30% of total building energy consumption and improved indoor air quality can be achieved by simply applying automated fault detection and diagnosis (AFDD), followed by corrections even if these are not done in real time.

Extensive research has been made on the development of component level AFDD tools in the past two decades. However, in recent years, the development of method for a whole building level AFDD has received much attention. Here, a whole building fault refers to a fault happening in one subsystem but impact on more than one subsystems, or has a significant impact on building performances, such as building's energy consumption. Existing component level AFDD tools could fail to detect or give false alarm for faults that trigger abnormalities in multiple subsystems. Isolating such whole building faults, which have impacts on multiple subsystems, is also challenging by simply using component based AFDD solutions. On the other hand, it might be more cost-effective for medium- or small-sized commercial buildings that have subsystems (e.g. primary cooling subsystem, air loop subsystem, heating subsystem, lighting subsystem, etc.) to install one whole building level AFDD tool, rather than many component level AFDD tools, to focus on significant faults.

In recent years, whole building energy simulation models have been employed for whole building AFDD. These whole building AFDD methods require the existence and calibration of a detailed whole building energy model, hence face challenges such as high development cost and low scalability. In the component AFDD field, both data-driven methods and expert knowledge/rule based methods have been successfully developed and adopted in the field. Nevertheless, when applying these conventional data-driven and expert knowledge/rule based methods for detecting and diagnosis faults that have a whole building impact, these methods encounter significant challenges such as 1) the curse of dimensionality, 2) difficulty to generate whole building level baseline, and 3) developing whole building level data-driven models. Moreover, similar to component level fault detection and diagnosis, developing whole building level AFDD solutions that are cost-effective, is always challenging. Moreover, there is generally a lack of training opportunity for undergraduate students to be engaged in research activities, especially in a multidisciplinary area such as AFDD research.

Therefore, the focus of this project is firstly to provide training opportunities in building AFDD research area for undergraduate students and to develop a workforce pipeline in building sciences. The objective of the proposed research activities is to develop cost-effective and scalable AFDD solutions for whole building level faults. The developed AFDD solution is VOLTTRON™ compatible to further increase its plug-n-play capacity and the market penetration. VOLTTRON™ is an open source innovative distributed control and sensing software platform developed by the Pacific Northwest National Laboratory. A VOLTTRON™ compatible AFDD tool brings the benefits such as low cost deployment, and better building to grid integration, etc. The performance

of the developed whole building AFDD tools is evaluated using data collected from a mixed use medium-sized university campus building. Another significant goal of this project is to engage undergraduate students in research activities and expose them to building control and AFDD fields.

More specifically, the following tasks have been successfully conducted in this project:

1) Undergraduate students from different disciplines have been engaged to develop various capabilities and professional skills in building sciences.
2) An extensive literature review is performed on a) existing AFDD methods developed for whole building fault, and b) those AFDD methods developed for other applications but have the potential to be used for whole building faults. This literature review helps to better understand current gaps and limits in the whole building AFDD area, and to explore potential solutions.
3) A demonstration building – Nesbitt Hall at Drexel University is identified for this study. This demonstration building has a typical HVAC system configuration, which includes a water cooled chiller system, three variable air volume (VAV) air handling unit (AHU) systems, and a hydronic heating system. Faults, that are expected to have a whole building level impact, are artificially implemented in this demonstration building for three different seasons. Building data have been collected in this project, which include fault free building automation system (BAS) data and BAS data that contain system behaviors caused by artificially implemented and naturally occurred faults. These collected data are used to evaluate the developed whole building AFDD tools.
4) A data-driven method, which includes a weather and schedule based pattern matching (WPM) method and Feature based Principal Component Analysis (FPCA) method, is developed for whole building level fault detection. Parameter sensitivity tests, including snapshot window size, data sample searching pool size, etc., are implemented to evaluate the impacts of various WPM-FPCA parameters.
5) A data-driven and expert knowledge/rule based method using Bayesian Network (BN) and WPM is developed for whole building level fault diagnosis. The developed WPM-BN method, which includes a two-layer BN structure model and BN parameters, are either learned from baseline data or developed from expert knowledge.
6) The developed whole building AFDD tools are evaluated using BAS data collected from the demonstration building. In total, there are eleven (11) whole building fault test cases (representing seven (7) types of whole building faults) selected from twenty five (25) fault test cases in which artificial faults have been implemented. Other fault test cases are not selected, due to the following reasons: 1) a fault has not been implemented correctly; 2) the implemented fault has not caused significant fault impacts due to certain specific weather or internal load conditions; and 3) significant missing data presented in the collected data for this fault test case. Three (3) naturally occurred fault test cases, representing two types of faults that have a whole building impact, are also selected to form the evaluation data set. Fourteen (14) fault-free test cases representing three seasons are selected. These fourteen (14) fault test cases and fourteen (14) fault-free test cases form an evaluation data set and serve as the “ground truth” when evaluating the developed AFDD methods.

7) Using the evaluation data set, the WPM-FPCA fault detection method is evaluated under two different Principle Component (PC) retention rates, i.e., 0.65 retention rate and 0.95 retention rate. It is observed that when using 0.65 PC retention rate, the WPM-FPCA fault detection method achieves a 79% fault detection rate (this rate reaches to 85% when including the six make-up fault test cases discussed in the next bullet), while when using 0.95 PC retention rate, the WPM-FPCA fault detection method achieves 100% fault detection rate (this rate remains to be 100% when including the six make-up fault test cases). Under both PC retention rates, the WPM-FPCA fault detection method demonstrates 0% false alarm rates.
8) Using the evaluation data set, the developed WPM-BN fault diagnosis method has successfully diagnosed thirteen (13) faults out of the fourteen (14) fault test cases. The one mis-diagnosed case exhibits fewer fault symptoms in different subsystems, when compared with other fault test cases. Overall speaking, the developed whole building AFDD solutions demonstrate satisfactory fault detection/diagnosis accuracy with very low false alarm rate.
9) Seven make-up tests were implemented in the summer of 2018 because of a lack of field data that contains those faults in the past tests. All tests demonstrated various symptoms caused either by the implemented faults, or by the concurrent faults. Among them, one fault test was not included into the evaluation due to a lack of data sample to generate the WPM baseline. Therefore, six fault tests were used to evaluate the developed AFDD tool. All six test days were successfully flagged by the WPM-FPCA method to show the overall system operation performance was affected. Although a ground truth was not well-established due to concurrent faults found in the nearly all six test days, the ranked posterior probability of each fault cause are in agreement with the fault symptoms.
10) The developed AFDD tool is VOLTTRON™ compatible. Here, VOLTTRON™ platform is served as a middle ware to connect with a BAS and to obtain BAS data. This BAS data is sent to the AFDD tool via the VOLTTRON™ platform. The integration with VOLTTRON is as a proof-of-concept to demonstrate that the AFDD platform developed can be ported relatively easily into a control execution platform.
11) A survey and market study has been performed to understand the existing AFDD market, including its gaps and needs.

This report provides a detailed description of the above tasks: Chapter 1 presents the research background and literature review. Chapter 2 illustrates the demonstration building. Chapter 3 introduces how faults are implemented in and how data are collected from the demonstration building, as well as the data analysis process. Chapter 4 discusses the development process of the whole building fault detection method. Chapter 5 introduces the development process of the whole building fault diagnosis method. Method evaluation and analysis are given in the Chapter 6. Chapter 7 illustrates how the developed whole building AFDD tools are integrated with the VOLTTRON™ platform. Chapter 8 describes the makeup experiments implemented in the summer of 2018. Chapter 9 presents the market studies. Chapter 10 summarizes how undergraduate students are engaged during the project. Chapter 11 provides the conclusions drawn from this project and discusses future research directions.

Acronyms and Abbreviations

AFDD	Automated fault detection and diagnosis
AHU	Air handling unit
ANN	Artificial Neural Network
BAS	Building Automation System
BIM	Building information modeling
BN	Bayesian Network
CHW	Chilled water
CPT	Conditional probability table
CW	Cooling water
DM	Data mining
FDD	Fault detection and diagnosis
FPCA	Feature based principal component analysis
GA	Genetic Algorithm
GUI	Graphic user interface
HVAC	Heating, Ventilation, Air-conditioning
MA	Mixed air
ML	Machine learning
MSPC	Multivariate statistical process control
OA	Outdoor air
PCA	Principal component analysis
PLS	Partial least square
PLSR	Partial least square regression
PM	Pattern matching
PPA	Piecewise Aggregate Approximation
RA	Return air
RMSECV	Root-mean-square error of cross-validation
SAX	Symbolic Aggregate Approximation
SPE	Squared Prediction Error
SVD	Singular value decomposition
SVM	Support Vector Machine
WPM	Weather/schedule based pattern matching
WBF	Whole building fault
VAV	Variable air volume
VFD	Variable frequency driver
VPF	Variable primary flow

Table of Contents

ACKNOWLEDGMENTS i
DISCLAIMER ii
Executive Summary iii
Acronyms and Abbreviations vi
Table of Contents vii
List of Figures xii
List of Tables xvi
Chapter 1 Introduction 1
1.1 Background 1
1.2 Challenges for Whole Building FDD methods 1
1.3 Research Objects 2
1.4 Literature Review 3
1.4.1 Definition of Whole Building Faults and Classification of Building Faults 3
1.4.2 Definition of AFDD and Classification of AFDD Methods in Building Systems 4
1.4.3 Whole Building FDD Work Flow 5
1.4.4 Whole Building FDD Methods 5
1.4.4.1 Overview of Model Based Whole Building FDD 5
1.4.4.2 Overview of Data-driven Whole Building FDD 7
1.4.5 Data-driven Whole Building Fault Detection and Diagnosis – Detailed Methods 10
1.4.5.1 System Feature Extraction 10
1.4.5.2 Baseline Generation 11
1.4.5.3 Feature Comparison and Outlier Detection 12
1.4.5.4 Fault Diagnosis Methods 13
1.5 Summary 13
Chapter 2 Demonstration Building Description 15
2.1 Demonstration Building –Nesbitt Hall Overview 15
2.2 Overview of HVAC System 16
2.2.1 BAS in the Nesbitt Hall 16

2.2.2 Primary Cooling Subsystem 17
2.2.3 Supply Air Subsystem 17
2.2.4 Steam-to-hot-water Heat Exchanger Subsystem 19
2.3 Summary 21
Chapter 3 Whole Building Fault Data Generation and Collection 22
3.1 Overview of Whole Building Fault Experiments 22
3.2 Primary Cooling System Fault 23
3.2.1 Chilled Water Supply (CHWS) Temperature Sensor Bias Fault 23
3.2.2 Chilled Water Differential Pressure Sensor Positive Bias Fault 24
3.3 Supply Air System Fault 25
3.3.1 AHU Supply Air Temperature Sensor Bias Fault 25
3.3.2 AHU Outdoor Air Damper Stuck at a Higher than Normal Position Fault 27
3.3.3 AHU Cooling Coil Valve Stuck at a Higher than Normal Position Fault 28
3.4 Operator Fault 28
3.4.1 AHU Cooling Coil Valve Software Operator Override Fault 29
3.4.2 System Scheduling Fault 29
3.5 Fault Implementation Consideration 30
3.6 Fault Implementation Methods 30
3.7 Summary 31
Chapter 4 Development of Fault Detection Method 32
4.1 Introduction 32
4.2 Overview of the WPM-FPCA Fault Detection Method 32
4.3 Development of Feature Selection Process 33
4.3.1 Introduction 33
4.3.1.1 Purpose of Variable Dimensionality Reduction 33
4.3.1.2 Category for Feature Selection Method 34
4.3.1.3 Basic Feature Selection Flow 35
4.3.2 Partial Least Square Regression (PLSR) 35
4.3.2.1 Introduction of PLSR 35
4.3.3.2 Wrapper Method in PLSR for Feature Selection 36
4.3.3 GA Searching 37

4.3.4 PLSR-GA based Feature Selection 38
4.3.4.1 Flow of PLSR-GA based Feature Selection 38
4.3.4.2 Cross Validation 38
4.3.4.3 Selection of GA Parameters 39
4.3.5 Variable Selection Result 40
4.4 Development of Weather/Schedule Information based Pattern Matching 42
4.4.1 Introduction of Data Pattern Matching 42
4.4.2 SAX Method 42
4.4.3 SAX Time Series Data Pattern Matching 43
4.5 Development of PCA for Building Fault Detection 45
4.5.1 Introduction 45
4.5.2 PCA Method 47
4.5.3 Statistics for PCA Fault Detection 49
4.5.4 Flow of PCA Fault Detection 50
4.6 Parameter Sensitivity Test for WPM-FPCA Fault Detection 50
4.6.1 Sensitivity Test on Snapshot Window Size 51
4.6.2 Sensitivity Test on Baseline Data Size and Sample Searching Pool Size 52
4.7 Summary 56
Chapter 5 Development of Fault Diagnosis Method 58
5.1 Introduction 58
5.2 Bayes Theorem 58
5.3 Development of BN Structure Model 59
5.3.1 BN Structure 59
5.3.2 BN Node 59
5.3.3 Development of BN Structure for Whole Building Fault 60
5.4 Development of BN Parameter 63
5.4.1 Development of Prior Probability Distribution 64
5.4.2 Development of Conditional Probability Distribution 65
5.4.3 Development of *LEAK* Probability Distribution 66
5.5 Fault Isolation Rule 68
5.6 Summary 69

Chapter 6 Method Evaluation 70
6.1 Ground Truth Establishment 70
6.1.1 Introduction 70
6.1.2 Method of Ground Truth Establishment 70
6.1.3 Whole Building Fault Identification and Fault Free Day Generation 71
6.2 Evaluation of Fault Detection 71
6.2.1 Fault Detection 71
6.2.2 Successful Detection Case Examples 72
6.2.2.1 AHU Outdoor Air Damper Stuck Fault at a Higher than Normal Position Fault 72
6.2.2.2 Chiller Stop Fault (Naturally Occurred Fault) 75
6.2.3 Mis-detection Case Example 76
6.2.4 Evaluation on False Alarm Rate 78
6.3 Evaluation of Diagnosis Method 81
6.3.1 Successful Diagnosis Case Example 82
6.3.2 Mis-Diagnosis Case Example 84
6.4 Evaluation on Method Implementation Cost and Payback 85
6.5 Summary 85
Chapter 7 Development of VOLTTRON™ Compatible AFDD Tool 87
7.1 Introduction 87
7.2 Development of VOLTTRON™ AFDD Agent 87
7.3 Development of AFDD GUI 88
7.3.1 Introduction of Microsoft Visual C#® 88
7.3.2 VC# based GUI Design 89
7.3.3 Web Browser-based GUI Design 90
7.4 Summary 92
Chapter 8 Makeup Fault Test 94
8.1 Introduction 94
8.2 Fault Tests and Method Evaluation 94
8.2.1 Fault Test Summary 94
8.2.2 Fault Test Description and FDD Result 94

8.2.2.1 Fault Test on July 9th 2018 97
8.2.2.2 Fault Tests on July 10th and July 18th 2018 100
8.2.2.3 Fault Test on July 11th, 2018 103
8.2.2.4 Fault Test on July 22nd, 2018 104
8.2.2.5 Fault Test on July 23rd, 2018 106
8.3 Summary 108
Chapter 9 Market Study 110
9.1 Market and Competition 110
9.2 Distribution Recommendations 110
9.3 Survey Study 111
Chapter 10 Undergraduate Engagement 112
Chapter 11 Conclusions and Recommendations 113
11.1 Conclusions 113
11.2 Recommendations 114
11.2.1 Recommendation on More Evaluation using Real Building Testbeds 114
11.2.2 Recommendation on Improvement of WPM-FPCA Fault Detection Method 115
11.2.3 Recommendation of the WPM-BN Fault Diagnosis Method 115
11.2.4 Recommendation on Baseline Data Collection 115
References 117
Appendix I Fault cases in Year 2016 and Year 2017 1
Appendix II List of Whole Building AFDD Tool Code 3
Appendix III List of VOLTTRON™ and BAS Interface Code 5
Appendix IV Market Survey Instrument 6
Appendix V Whole Building Fault Energy Impacts 14
Appendix VI Variable Selection Result 15

List of Figures

Figure 1-1 Classification scheme for FDD methods 4
Figure 1-2 Top-down and bottom-up methods used for HVAC FDD 6
Figure 1-3 The procedure of detecting abnormal energy consumption 9
Figure 1-4 Basic data-driven FDD procedure 9
Figure 2-1 Overview of the Nesbitt Hall at Drexel University 15
Figure 2-2 HVAC system configuration in the Nesbitt Hall 16
Figure 2-3 BAS graphic interface 17
Figure 2-4 VPF primary cooling sub-system configuration in the Nesbitt Hall 18
Figure 2-5 Centrifugal chiller in the Nesbitt Hall 18
Figure 2-6 Cooling tower configuration in the Nesbitt Hall 19
Figure 2-7 AHU with return air duct 19
Figure 2-8 AHU without return air duct 20
Figure 2-9 AHU-1 profile 20
Figure 2-10 VAV terminal unit 21
Figure 2-11 Heat exchangers 21
Figure 3-1 CHWS temperature setpoint configuration on chiller control panel 24
Figure 3-2 Chilled water supply differential pressure sensor logic configuration interface .. 25
Figure 3-3 Chilled water differential pressure setpoint configuration 25
Figure 3-4 AHU supply temperature setpoint configuration 26
Figure 3-5 AHU economizer adjustment 27
Figure 3-6 AHU cooling coil valve adjustment 28
Figure 3-7 System scheduling configuration 29
Figure 3-8 Fault test implementation flowchart 30
Figure 4-1 Architecture of WPM-FPCA fault detection method 33
Figure 4-2 Categories of dimension reduction method 35
Figure 4-3 Illustration of wrapper method for feature selection 37
Figure 4-4 Flow of GA for feature selection 39
Figure 4-5 Variable selection result from PLSR-GA method 41
Figure 4-6 method key variable selection result 42

Figure 4-7 Illustration of PPA method 43
Figure 4-8 SAX word in tagging enthalpy condition 45
Figure 4-9 Flow of WPM by using SAX 46
Figure 4-10 Illustration of WPM by using SAX 46
Figure 4-11 Decomposition of measurement matrix X 47
Figure 4-12 Number of PCs determination by using scree plot 49
Figure 4-13 Flowchart of WPM-FPCA for fault detection 50
Figure 4-14 PM comparison under different window sizes 51
Figure 4-15 Fault detection comparison under different window sizes 52
Figure 4-16 Fault detection comparison under different window sizes 52
Figure 4-17 Illustration of data searching pool 54
Figure 4-18 Frequency of number of data sample for the 30 test days in summer season 55
Figure 4-19 Frequency of number of data sample for the 30 test days in fall season 56
Figure 4-20 Frequency of number of data sample for the 30 test days in winter season 56
Figure 5-1 BN structure model example 60
Figure 5-2 BN structure for whole building fault diagnosis 62
Figure 5-3 Illustration of Noisy-OR gate 64
Figure 5-4 *LEAK* probability (chilled water flowrate) 66
Figure 5-5 *LEAK* probability (chilled water return temperature) 67
Figure 5-6 *LEAK* probability (chilled water supply temperature) 67
Figure 5-7 *LEAK* probability (chiller cooling energy) 68
Figure 6-1 Fault detection result under 0.95 PC retention rate (July 11th, 2017) 72
Figure 6-2 Chiller plant operation status (July 9th, 2017) 75
Figure 6-3 Fault detection result under 0.95 PC retention rate (July 9th, 2017) 76
Figure 6-4 AHU-1 and AHU-2 cooling coil operation status (July 9th, 2017) 76
Figure 6-5 VAV terminal unit operation status (July 9th, 2017) 77
Figure 6-6 Chiller operation status (August 3rd, 2017) 77
Figure 6-7 AHU-1 operation status (July 9th, 2017) 78
Figure 6-8 Fault detection result (August 3rd, 2017) under 0.65 PC retention rate 79
Figure 6-9 Fault detection result (August 3rd, 2017) under 0.95 PC retention rate 79
Figure 6-10 Detection result in August 28th, 2016 (summer season) 80

Figure 6-11 Detection result in Dec. 8th, 2016 (transition season)..... 81
Figure 6-12 Detection result in Jan. 24th, 2017 (winter season)..... 81
Figure 6-13 Fault detection and diagnosis result (July 11th, 2017)..... 82
Figure 6-14 Fault diagnosis result (August 3rd , 2017) 84
Figure 7-1 Software structure..... 87
Figure 7-2 AFDD agent working flow..... 88
Figure 7-3 Login interface 89
Figure 7-4 Fault detection result interface 89
Figure 7-5 Fault diganosis result interface..... 90
Figure 7-6 VOLTTRON™ connection interface..... 90
Figure 7-7 Login screen of web-based AFDD demonstration platform 91
Figure 7-8 Web-based fault detection interface..... 91
Figure 7-9 Web-based fault diagnosis interface..... 92
Figure 7-10 Key measurements 93
Figure 8-1 AHU-2 operation between the test day and the baseline (July 9th, 2018) 97
Figure 8-2 Chiller plant operation between the test day and the baseline (July 9th, 2018) 98
Figure 8-3 AHU-1 operation between the test day and the baseline (July 9th, 2018) 98
Figure 8-4 Fault detection result under 0.95 PC retention rate (July 9th, 2018)..... 99
Figure 8-5 Fault diagnosis result (July 9th, 2018) 99
Figure 8-6 AHU-1 operation between the test day and the baseline (July 10th, 2018) 100
Figure 8-7 AHU-1 operation between the test day and the baseline (July 10th, 2018) 101
Figure 8-8 Fault detection result under 0.95 PC retention rate (July 10th, 2018)..... 102
Figure 8-9 Fault detection result under 0.95 PC retention rate (July 18th, 2018)..... 102
Figure 8-10 Fault diagnosis result (July 10th, 2018) 103
Figure 8-11 Fault diagnosis result (July 18th, 2018) 103
Figure 8-12 AHU-2 operation between the test day and the baseline (July 11th, 2018) 104
Figure 8-13 AHU-1 operation between the test day and the baseline (July 11th, 2018) 105
Figure 8-14 Fault detection result under 0.95 PC retention rate (July 11th, 2018)..... 105
Figure 8-15 Fault diagnosis result (July 11th, 2018) 106
Figure 8-16 Fault detection result under 0.95 PC retention rate (July 22nd, 2018)..... 106
Figure 8-17 Fault diagnosis result (July 22nd, 2018)..... 107

Figure 8-18 AHU-1 operation between the test day and the baseline (July 23rd, 2018) 107
Figure 8-19 AHU-2 operation between the test day and the baseline (July 23rd, 2018) 108
Figure 8-20 Fault detection result under 0.95 PC retention rate (July 23rd, 2018) 109
Figure 8-21 Fault diagnosis result (July 23rd, 2018) 109

List of Tables

Table 2-1 BAS connected subsystem or measurement summary 16
Table 3-1 Whole building faults implemented or natrually occurred in this research..... 22
Table 3-2 Primary cooling system fault tests 23
Table 3-3 Supply air fault tests 26
Table 3-4 Operator fault test 29
Table 4-1 PLSR-GA parameter setting 40
Table 4-2 Sensitivity analysis on fault free date searching number..... 53
Table 4-3 Sensitivity test on data searching pool size 55
Table 5-1 Whole building fault list 61
Table 5-2 Fault evidence list 61
Table 5-3 Fault and evidence list 62
Table 5-4 Prior Probability for fault node 65
Table 5-5 CPT for evidence node 65
Table 5-6 *LEAK* probabilities for each measurement 68
Table 6-1 Key BAS measurements and abnormality threshold 71
Table 6-2 Whole building fault identification (Cases from 2016 to 2017)..... 73
Table 6-3 Fault detection result (Cases from 2016 to 2017)..... 74
Table 6-4 False alarm analysis 80
Table 6-5 Fault diagnosis result (Cases from 2016 to 2017) 83
Table 8-1 Fault test and FDD results 95
Table 10-1 Undergraduate Engagement..... 112

Chapter 1 Introduction

1.1 Background

Commercial and residential building sectors consumed 40.1% of the U.S. primary energy and represented 40% of the carbon emission in 2011[1]. Malfunctioning control, operation, and building equipment are considered as the top cause for "deficient" building systems [2, 3]. These dramatically increase the energy consumption (estimated to be $1x10^{15}$ BTU for commercial building primary energy usage [3] and between 0.35 to 17 quads of additional energy consumption caused by key faults at a national level [4]), and significantly impact the health/productivity of occupants. Field studies have shown that an energy saving of 5-30% and improved indoor air quality can be achieved by simply applying automated fault detection and diagnosis (AFDD) followed by corrections even if these are not done in real time [5, 6].

Existing AFDD solutions mainly focus on component level detection and diagnosis, decoupling the connections between the building subsystems – consequently, these may reach local and often incorrect solutions, yet they may not lead to overall sustainable and optimally conditioned systems. For example, when the outdoor air damper of an air handling unit (AHU) is stuck at a position that is much larger than normal in a cooling season, this fault not only could cause the AHU's cooling coil valve to be opened at higher than normal position, but also could cause abnormalities in the primary cooling system, such as causing the chilled water pump to run at a higher speed to provide extra cooling needed. A false alarm may be triggered by the primary cooling system AFDD tool package. In this study, such faults that occur in one equipment or component but would have fault impacts (causing abnormalities) on more than one component and subsystem, or would have significant adverse whole building energy or indoor environment impact, are considered as whole building faults. Besides having difficulty isolating whole building faults, existing AFDD solutions often encounter high market barriers because they have limited scalability, or high implementation cost due to the needs to manually customize algorithms, collect data on the field, especially faulty data. Moreover, many of the reported research are based on simulated system data which could be very different from real system data, and do not reflect real system's behavior. Therefore, more real building AFDD testing and validation are desired to evaluate the performance of an AFDD method.

It is noticed that existing literature does not clearly separate the concepts of AFDD and fault detection and diagnosis (FDD). A reason for this could be due to the challenge to define the concept of "automated", as many AFDD strategies may still engage some manual efforts. Hence these two concepts are not strictly separated in this report, although we focus on the methods and strategies that lead to a higher level of automation in FDD.

1.2 Challenges for Whole Building FDD methods

There exist five major challenges in developing whole building AFDD methods:

1) Challenge of cost-effectiveness and scalability

Although some model-based whole building AFDD tools have been developed and they can potentially provide accurate fault detection and diagnosis, developing such models and calibrating such models to reach an accuracy level that is sufficient for whole building FDD remains to be

challenging and cost-effective. At the same time, such tools often have lower scalability when applying in different buildings, due to the needs to customize them.

2) Challenge of the curse of data dimensionality

Different from component level AFDD, a whole building AFDD tool needs to utilize data from the entire building [7], which often has hundreds, if not thousands of data points. How to handle such an increased data dimensionality in a computationally efficient way, also referred to as the curse of the data dimensionality, is a challenge.

3) Challenge of generating baseline data that are under identical weather and internal load

For building systems, it is often challenging to differentiate weather and/or internal load triggered system operational changes from fault triggered abnormalities. Difficulties exist as to how to collect historical baseline data, and how to efficiently select among these collected historical baseline data, those baseline data that are under similar weather and internal load conditions as the snap-shot data (data from current system operation). Notice that a real building system is often impossible to be at a true "fault free" status. In this study, we use the term "fault free" to reflect a status at which a building system's performance is acceptable, such as when a building is just commissioned. Building data collected under such "fault free" status is referred to as baseline data.

4) Challenge of root cause identification

Although limited whole building AFDD studies exist in the literature, these studies often focus on fault detection, i.e., identify abnormalities. There is a lack of studies in the literature that focus on locating and identifying root causes for those faults that cause abnormalities in multiple subsystems/components, i.e., have coupled abnormalities.

5) Challenge of real building evaluation

Literature-reported FDD methods are mostly developed and evaluated using simulated system data. This is due to the difficulties of obtaining and analyzing real building data. Implementing faults and obtaining data that contain fault impacts in real buildings are already challenging. Yet cleaning and analyzing building data to obtain "ground truth" is even more arduous since unexpected naturally occurred faults could exist in the system and cause abnormalities or complicate (sometimes even eliminate) the fault impacts expected from the artificially implemented faults.

1.3 Research Objects

The main objectives of this project are to 1) develop a cost-effective and scalable AFDD tool that integrates both statistical process control, machine learning methods and rule-based methods to achieve a whole building system root-fault diagnosis and 2) evaluate the performance of the developed tools using real building data. Besides, the developed tools will be VOLTTRON™ compatible to further increase its plug-n-play capability and market penetration. VOLTTRON™ is an open source innovative distributed control and sensing software platform developed by the Pacific Northwest National Lab (PNNL, http://gridoptics.pnnl.gov/VOLTTRON/). Another objective of the project is to engage undergraduate students in research activities to encourage them to enter the sustainable building field. A university building is used as a demonstration site

for this project, which will further be used as an educational tool to train future architectural and mechanical engineers the significance and knowledge base of fault diagnosis. The primary final products are software tools (codes and data/user interfaces) for cost-effective and accurate whole building root-fault detection and diagnosis.

1.4 Literature Review

1.4.1 Definition of Whole Building Faults and Classification of Building Faults

A definition for whole building faults are not found in the previous literature to the author's best knowledge. In this research, a whole building fault refers to a fault that occurs in one equipment or component but has fault impacts (abnormalities) on more than one component and subsystem, or has significant adverse impacts on whole building level energy consumption and/or indoor environment quality.

Although, it is well-recognized that faults can lead to occupants' discomfort, system performance degradation, lower system life time and increased energy usage, there is a handful of investigation on typical fault types and their impacts, especially their impacts on coupled systems. In [8], many typical HVAC faults are identified according to their impacts on the energy consumption. Furthermore, energy impacts from AHU [9, 10] and variable air volume (VAV) system [11] were also studied. A very complete category for AHU faults were reported in [9, 12]. However, a systematic fault categorization for whole building faults is not found.

Following fault categorization methods in other industrial systems, HVAC faults, including whole building faults, can be categorized according to the fault types, fault duration, fault source location, and fault causes.

Based on the fault types, faults can be divided into hardware faults and software faults [13]. Hardware fault refers to the faults happened in the hardware system including sensor, actuators, controllers, energy or air or fluid distribution equipment and so on. Software fault refers to the faults caused by software system or operator. For example, a software program fault can lead the system operation abnormality. Moreover, some incorrect system software settings (i.e., a wrong tuning of PID parameters) can also be categorized as software faults.

According to fault duration, faults can be identified as permanent faults or nonpermanent faults [14]. A permanent faults usually is a long term fault and will not disappear until the system is serviced and repaired [14]. For example, in building systems, a duck leaking fault is a permanent fault. Unobserved permanent faults can cause a long-lasting system deterioration and may lead to a complete system failure or an accumulated energy waste in a HVAC system. Nonpermanent faults include intermittent faults and transient faults. Intermittent faults occur repeatedly during a time period [14]. For example, an incorrect occupancy schedule accidently set by the operator can lead the building systems to be operated during unoccupied hours and cause energy waste. Transient faults occur randomly and are usually unrepeatable. For example, an AHU outdoor air damper stuck fault could happen randomly and could also return back to a normal position without manual correction. In such case, fault impacts may not be observed at all times during system operation.

As building's systems are often large and complex in which different subsystems are coupled together, the source location of a fault, rather than its impacts, is better to be utilized to categorize

whole building faults. Hence, in this research, whole building faults can be identified as operator faults, primary heating/cooling system faults, and supply air/terminal system faults. An example of detailed whole building fault category is provided in Chapter 3.

1.4.2 Definition of AFDD and Classification of AFDD Methods in Building Systems

For AFDD in building systems, Haves [15] provides good definitions. In this paper, fault detection is defined as "determination that the operation of the building is incorrect or unacceptable in some respect" and fault diagnosis is defined as "identification or localization of the cause of faulty operation". Compared to fault detection, fault diagnosis is much more difficult because more knowledge, information, and analysis are needed to isolate the root causes which impact the system performance.

However, although there is no specific definition for whole building AFDD, the above wordings for FDD are still applicable for whole building level faults. Here we define the term whole building AFDD as the detection and diagnosis of a whole building fault.

Many methods have been developed for component level AFDD for building systems. In [6], Katipamula and Brambly provided a comprehensive classification for methods used for HVAC system AFDD as shown in

Figure 1-1. This taxonomy can be used to determine the methods employed in the whole building AFDD as well. In the following sections, we group quantitative model-based methods and qualitative model-based methods into model-based methods. We classify the process history based methods into data-driven methods.

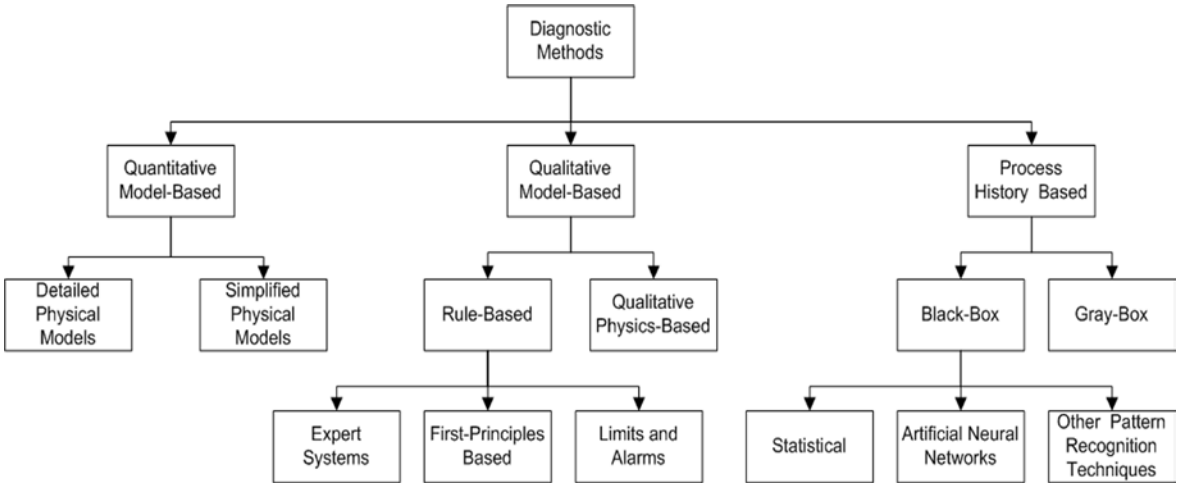


Figure 1-1 Classification scheme for FDD methods [22]

In the middle of 1990s, the DOE started to develop a whole building detection and diagnosis tool [16]. Some building commissioning software such as Performance and Continuous Recommissioning Analysis Tool developed by Facility Dynamics Engineering [17] and Continuous Commission® tool [18] were developed to preliminarily evaluate the overall building performance. Nevertheless, it is noticed that the research work on the whole building AFDD was not as active as those in the component AFDD area, before Year 2000. Five academic papers were found between Year 2000 to Year 2010 in the whole building AFDD area. After Year 2010,

relatively more research work on the whole building FDD area had been published. The following sections provide a more detailed discussion on the whole building AFDD studies published after Year 2000.

1.4.3 Whole Building FDD Work Flow

Existing literature suggests that there are two typical AFDD strategies at the whole building level—top-down strategy and bottom-up strategy [19]. Figure 1-2 illustrates the differences between these two by using the HVAC system as an example [20].

The top-down approach is defined as measuring the whole performance abnormality at higher system levels and then reason the causes of faults, mal-functioning or degradation at lower-levels. Building energy consumption is usually selected as the fault detection index because of its simplicity and significance. However, most existing studies have skipped the root cause diagnosis step and stop at the phase of detecting the abnormal energy consumption. Only limited studies [21-23] completed the entire top-down approach to identify the fault sources. Wu et al., diagnosed the faults such as thermostat setpoint variation, VAV outlet blockage in [21], and faults such as AHU fan frozen, outdoor air temperature sensor malfunction, VAV supply air inlet blockage [22]. Seem et al., [23] used field data to detect abnormalities caused by faults including chiller failure, design deficiency, and improper operation.

The bottom-up approach measures lower level system parameters and back propagates the fault information to a higher level to evaluate the fault's impacts on whole building performance. In this approach, component or equipment fault are detected first. The cumulative fault impacts are then evaluated after the faults are detected and diagnosed. The fault detection and diagnosis process is the same as component level FDD strategy. Studies [24-27] followed this approach, among which, three studies [24, 25, 27] did not start the detection work flow from the component level but from the subsystem level. Furthermore, these three studies did not diagnose the root fault causes, nor did they provide detailed evaluation of the fault impacts on the whole building level. Du et. al. [26] presented a FDD method that started from detecting the component level faults including water loop return water temperature sensor fault, supply air temperature sensor fault, and chilled water valve fault. But it did not provide a detailed analysis on the impacts of a specific fault on multiple systems. The challenge of such bottom-up approach is that certain faults could cause malfunctions on multiple systems, such as the outdoor air damper fault discussed earlier. Hence, based on existing literature, component or subsystem level FDD faces challenges when isolating root fault causes if an abnormality is caused by a fault outside of this component/subsystem.

1.4.4 Whole Building FDD Methods

Similar to component FDD methods as shown in Figure 1-2 [28, 29], whole building FDD methods is divided into model-based methods and data-driven methods. This section provides a discussion of the whole building FDD methods reported in the literature.

1.4.4.1 Overview of Model Based Whole Building FDD

There are generally two model-based FDD methods—quantitative model-based and qualitative model-based methods (Figure 1-2) [30]. Quantitative model-based methods use detailed physical models or simplified models of the physical processes [6]. Qualitative model-based methods consist of the basic qualitative description of the system dynamic process. This

description, which is based on the qualitative relationships or prior knowledge, is then used for fault detection and diagnosis. Rule-based method is usually considered as the qualitative method.

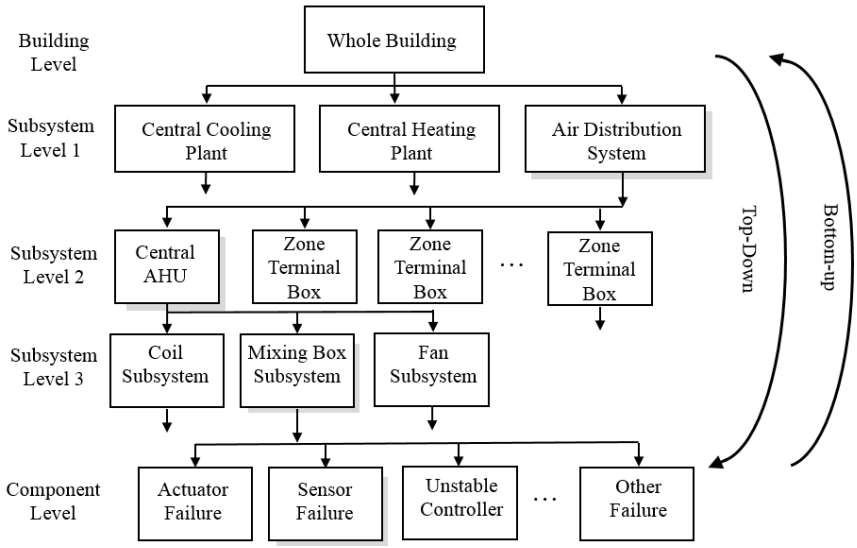


Figure 1-2 Top-down and bottom-up methods used for HVAC FDD [19]

In the whole building FDD area, detailed physics based building energy consumption models, such as those built using EnergyPlus [31], HVACSIM$^+$ [32] and TRNSYS [33], had been adopted to produce baseline energy consumption, which was forecasted from the energy consumption of each component. Real building measurements from the BAS could be used to calibrate such whole building energy consumption models.

Haves et al., [34] described a basic modeling and implementation procedure for using detailed whole building energy simulation in on-line performance assessment. The following three studies [35-37] adopted such detailed building energy simulation approaches for whole building fault detection.

Bynum et al., [35] developed an Automated Building Commissioning Analysis Tool, which was a simulation tool combining expert system to detect energy abnormality. A building energy model was firstly developed by using first principles. Actual energy consumption data collected from the BAS combining with weather data were used to calibrate the simulation model. Forecasted energy performances from the calibrated simulation model were then compared with the measured data to detect faults. Five buildings were used to demonstrate the detection results. In this paper, the authors did not describe how the faults were implemented in the demo buildings (or whether they were naturally occurred faults). The fault detection accuracy and detection threshold were not described either.

O'Neill et al., [36] described another model-based FDD method. In their approach, energy consumption data from different subsystems including HVAC, lighting and plug equipment usage were compared with a reference EnergyPlus model which was used to calculate the annual energy consumption to determine if the system had operation faults. A diagnosis process, which utilized the system hierarchy and the model vs. measurements deviation in each subsystem, was also described.

In article [37], Dong et al., presented a Building Information Management System (BIMS) based on Autodesk Revit, which was used to serve as the platform for a baseline energy performance model developed by MATLAB®. Building Control Virtual Test Bed developed by Laurence Berkley National Laboratory was used to collect real building energy data. Again, the differences between the real measurements and the forecasted energy performances from the BIMS model were used to detect potential faults. In this paper, a real building which had AHU and connected VAV were used as the case study to demonstrate the method performance. It is noted that, the three papers described above did not provide detailed descriptions on 1) how to decide on the threshold for abnormality; nor 2) detection/diagnosis accuracy.

The advantage of model-based FDD methods using detailed building energy simulation is that these methods could provide a relatively accurate energy baseline (if the simulation models are well calibrated) to analyze the building's energy performance. If the developed model accurately represents subsystems and whole building level operation, then fault diagnosis is not too difficult because both subsystem level and whole building level comparison (model vs. measurements) can be achieved to isolate faults. However, the main disadvantage of this approach is that in practice, developing such a detailed and accurate building energy simulation model is very time-consuming, and calibrating such a model to achieve a subsystem level accuracy for fault isolation remains to be very challenging.

1.4.4.2 Overview of Data-driven Whole Building FDD

Although model-based methods, including those using detailed energy simulation models, can be successful for both component and whole building level FDD, it is challenging to cost-effectively implement such methods in the real practice, especially for whole building FDD. Besides model-based methods, data-driven FDD methods have also been successfully employed in the component level FDD [38, 39]. Such data-driven FDD methods use multivariate statistics and machine learning strategies to detect and diagnose system operational, as well as control faults [40]. Moreover, data-driven FDD methods have demonstrated great potentials for whole building FDD, especially when system is too complex to be modeled, and when it is easy to obtain a large amount of measurements.

There are a few studies reported in the whole building FDD area using various data-driven methods. A typical implementation of data-driven methods for building energy abnormality detection is summarized in [23] and is illustrated in Figure 1-3. Here, we categorize and re-define these basic procedures into five main steps as shown in Figure 1-4. This categorization will help later discussion on data-driven methods, as a specific technique, such as pattern matching, can be used in different processes to achieve different objectives. It is noted that not existing studies strictly follow all of these five steps:

1. The first step is the data collection and data pre-processing process. In this step, system operation data is collected from the BAS. By now, this process is still the most time-consuming and labor intensive process due to the randomness of BAS data points naming strategy and the complexity/diversity of BAS database design. In existing literature, both real building data and simulated data (in lieu of real building data) have been used. Some studies [21-23, 25, 35-37, 41-43] use real data from the field test. Yet other studies [26, 27, 44, 45] use data from simulation model. Unlike in the chemical process industry, where a unified system process model such as the Tennessee Eastman process system model has

already been well-established [46, 47], in the building sector, there has been a lack of standardized building simulation model to validate a FDD method. Hence, it is difficult to evaluate an FDD method's accuracy and cost-effectiveness when different sources of building data are used. Moreover, data-driven methods are sensitive to data quality. It is challenging to evaluate how a data-driven FDD method would perform in a real building, even if it performs well for a simulated building. For a data-driven FDD method, the quality of the collected data strongly affect the FDD method's performance. For systems which have few information, the application of a data-driven method is greatly limited [36]. Although modern buildings often have BAS to collect data, 'data rich yet information poor' [48] phenomenon happens frequently in buildings. Unfortunately, existing literature has not discussed data source topics such as how to efficiently collect data and how to understand the impact of data uncertainty on FDD accuracy. This topic remains to be a challenge for future research.

2. The second step is the system feature extraction process. Considering the much larger quantity of building data, this is a key step for whole building FDD, compared to component level FDD. Hundreds to thousands of data points exist in a BAS. But many of such data are correlated or redundant. Being able to efficiently remove unusable and/or redundant data, and to identify useful information from raw data is the key for a whole building FDD. In existing whole building FDD literature, Support Vector Machine (SVM) [48], and other data mining techniques such Artificial Neural Network (ANN) [24, 26, 42], have been employed for this step. However, their accuracy and efficiency are not thoroughly discussed.
3. The third step is the establishment of a baseline. Again, a baseline refers to a system status, including corresponding system measurements, that is considered to be an acceptable (normal) operation status. A fault is identified when the system's performance demonstrates a large enough (higher than a predefined threshold) deviation from the baseline. Defining baseline and threshold remains to be a great challenge even for component level FDD, due to the large variety of system customization in a real building, and that a "fault-free" status never truly exists in a real building. Typically, a baseline is established based on predefined rules and extracted system information. Energy consumption data are the most commonly used measurements for whole building FDD baseline, which are used in all of the whole building FDD literature identified. Some studies establish the baseline using direct energy measurements from real buildings [25, 27, 45, 49]. Other studies establish the baseline based on forecasted data using correlation models [43]. Although building energy consumption is an easy-to-use baseline measurement to detect abnormality, it does not provide enough information for root fault isolation, which could be the reason that very few literature discusses fault diagnosis.
4. The fourth step is the feature comparison and fault (outlier) detection. In this step, the real-time BAS data are compared with the baseline to detect if the system has faults. Threshold is the key factor affecting detection effectiveness and accuracy. If the threshold is too narrow, it will cause a high false positive (detecting a fault when there is no fault) rate. If the threshold is too wide, it will cause a high false negative (not detecting a fault when there is a fault) rate. The threshold can be established by using system operational data and expert knowledge [25], or purely through statistical analysis [21].
5. The fourth step is the feature comparison and fault (outlier) detection. In this step, the real-time BAS data are compared with the baseline to detect if the system has faults. Threshold

is the key factor affecting detection effectiveness and accuracy. If the threshold is too narrow, it will cause a high false positive (detecting a fault when there is no fault) rate. If the threshold is too wide, it will cause a high false negative (not detecting a fault when there is a fault) rate. The threshold can be established by using system operational data and expert knowledge [25], or purely through statistical analysis [21].

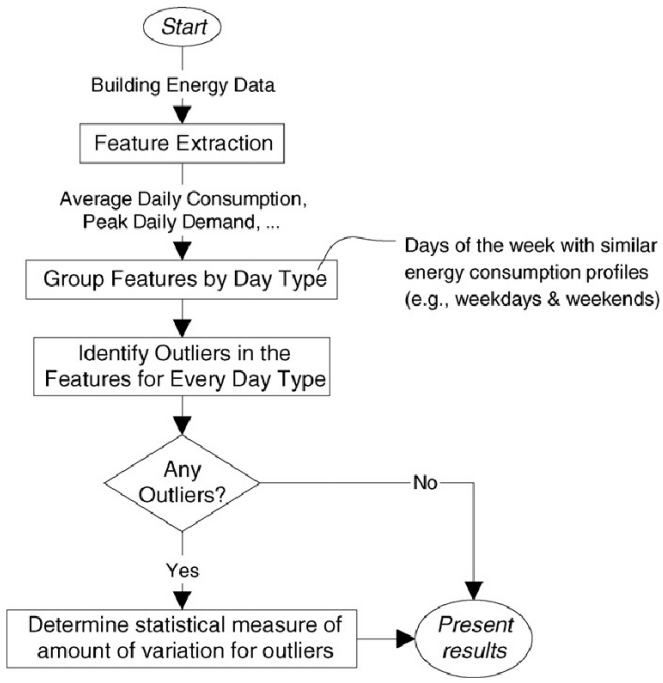


Figure 1-3 The procedure of detecting abnormal energy consumption [23]

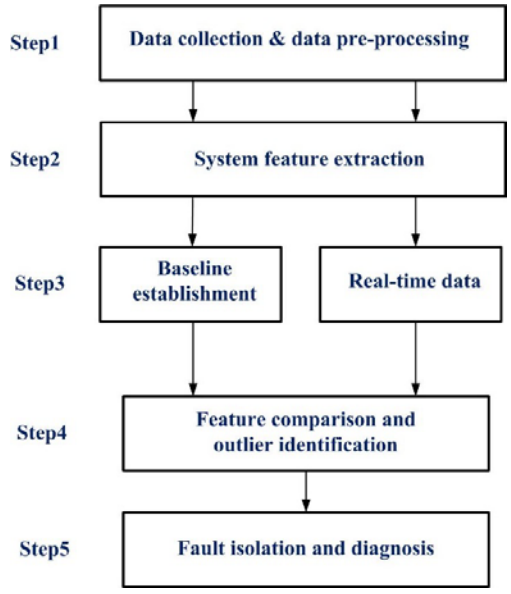


Figure 1-4 Basic data-driven FDD procedure

6. The last step is to diagnose the root fault cause. In this step, various statistic parameters or probability model such as ANN and Bayesian belief network reasoning method have been reported [21, 24, 26, 44]. More detailed discussion is provided in the next section.

1.4.5 Data-driven Whole Building Fault Detection and Diagnosis – Detailed Methods

In this section, methods that could potentially benefit data-driven whole building FDD are discussed here. The methods reviewed in this section include those reported for system feature extraction, baseline establishment and fault diagnosis. Detailed information including detection and diagnosis principles, types of data used, baseline establishment principles, and major detected faults are discussed.

1.4.5.1 System Feature Extraction

System feature extraction process is similar to the model development process in a model-based method. However, building operation data instead of first principles are used to represent the system in a system feature extraction process. ANN, SVM, data based energy flow model have been reported to extract system features.

Dodier et al., [24] proposed to use four Energy Consumption Indexes to establish a building energy consumption baseline, which included building total electric, building total thermal, non-chiller HVAC electric, and chiller/package unit electric. ANN models were established to predict the expected energy end-uses. Considering the large variation of system operation models and weather conditions, even for the same building, large quantity of training data would be needed for ANN model training. It was unclear whether the reported ANN model could be retrained for a new building, even if large quantity of training data were available from the new building. Besides, the paper did not provide details such as the training data duration and accuracy evaluation which would be important for the method to be used in a real practice.

Blanes [48] et al., proposed a “CASCAD implementation kit” in which a FDD tool was integrated into an energy management system. In this tool, SVM, which had the ability to reduce BAS data dimension, was used to extract building energy information. Again, there was a lack of details on how such algorithm was implemented and the accuracy of the method in this article.

Wu et al., [31,32] developed an energy flow model which was based on Bernoulli’s principle for all HVAC units. Energy consumptions by different HVAC units were treated as energy density. Operational patterns based on this energy density under normal conditions were served as the baseline. The energy consumption was analyzed by a method named as “temporal and spatial partition strategy”. The temporal partition method required a detailed analysis on the HVAC system unit placement according to their locations. The spatial partition method used different weather conditions to differentiate the unit energy consumption. However, the authors did not give a detailed analysis on how to differentiate energy consumption fluctuation due to faults vs. due to different operational modes, although theoretically, the temporal partition method should be able to help to differentiate operational modes. Moreover, faults from different components might cause similar impacts on the energy consumptions, which could weaken the ability of fault diagnosis. Moreover, how scalable this method was when being applied to a different system or a different building was not studied.

An efficient feature extraction method is critical for FDD, especially for whole building FDD, when much larger quantity of data exist. Overall speaking, existing research has reported various data-driven methods to extract system information and to reduce system complexity. However, there is a lack of discussion on critical details, such as how data quantity, quality, uncertainty etc. affecting the reported method's performance, and thus it is hard to evaluate these reported methods, especially their scalability.

1.4.5.2 Baseline Generation

After system information is extracted, a baseline needs to be established to represent normal operation. A threshold will then need to be defined to differentiate abnormal operation from normal operation for fault detection.

(1) Baseline generation

In the component level FDD, operational parameters measured by sensors or control signals are usually used to detect faults [46]. In the whole building FDD, as discussed earlier, energy analysis and measurements are easy-to-use parameters for baseline establishment. The complexity here is that building energy consumption is affected by many factors (predicting factors), such as internal loads, operating conditions and weather conditions. Various approaches to choose effective predicting factors are studied in the literature. Weather conditions including outdoor temperature and relative humidity are the mostly common ones to establish the baseline.

Lin et al., [25] introduced a temperature-based approach — Days Exceeding Threshold-Toa to detect abnormal building energy consumptions. In their study, the ASHRAE Simplified Energy Analysis Procedure was used to establish a simulation model. Building cooling and heating energy consumption in the baseline period, chosen from a post-commissioning period, were used to calibrate the model.

Miller et al., [41] presented a day-typing process (named as DayFilter) which employed Symbolic Aggregate Approximation (SAX) to extract the most common daily profiles. At the same time, day-typing motif and discord candidate data were created after analyzing the day feature. *K*-means based clustering algorithm was used to cluster the daily profiles to create the baseline. This method provided a fairly quick analysis to filter diurnal patterns of the building performance data.

Time series regression modeling is a commonly used method to establish a baseline. This method attempts to use various energy consumption regression models combined with weather information to establish the baseline. However, system degradation faults could be neglected when using time-series regression methods. Fan et al., [49] presented a time series based data mining methodology including SAX, motif discovery and temporal association rule mining for building temporal operation data. The dynamic characteristic of BAS data was analyzed to establish the operation pattern, derive temporal association rules and evaluate the system performance. Although this paper did not use the developed baseline for fault detection, efficiently using building energy consumption information will improve the FDD effectiveness.

Most data-driven FDD methods detect faults by comparing real time or near time operation data with baseline data obtained from historical data. The challenge is again finding the correct historical data that can represent normal operation, considering that fault free status rarely exists in a real building. Comparing to traditional data-driven methods, Li et al., [43] developed a different prediction methods to detect the whole building energy abnormality. In their study, they firstly extracted building energy features from time-series BAS data, and then employed the

features to detect abnormal energy consumptions. A canonical variate analysis was used to describe latent variables of building daily electricity consumption profiles, and then group the data sets into different clusters. The classification was produced to predict daily electricity consumption. Comparing with traditional regression energy forecasting models, this method had a potential to build a more robust baseline by removing data outliers and infrequent energy abnormality that could exist in historical data. The algorithm also had higher computational efficiency. However, detailed performance of the method, such as accuracy, was not discussed in this article.

(2) Identification of threshold

Threshold is a critical factor that is used to detect faults quantitatively [42]. How well a threshold is chosen also strongly affects an FDD method's detection accuracy and false alarm rate. Two methods have been discussed in the whole building FDD literature to establish a threshold. One method is to directly use system operational information such as consumption variation to determine the threshold. Lin and Claridge [25] used hot water and chilled water consumption variation to establish threshold. However, in this study, how to determine an exact numerical value for the threshold was not discussed for each test case. Another method is to use statistical parameters to determine a threshold. Wu et al., [21] set 95% probability interval as the threshold for fault detection when using Principal Component Analysis (PCA) method.

It is noted that, despite its significance, very few existing literature in the whole building FDD area have discussed how to determination a threshold nor have reported their threshold values.

1.4.5.3 Feature Comparison and Outlier Detection

Feature comparison and outlier detection help to detect system abnormal behavior. In this stage, different thresholds are set up to determine and detect outliers. Although, this stage is actually based on the work in previous steps, we can find some literature that provides details or innovative methods to implement outlier detection process.

Seem [23] used a generalized extreme student's deviate for outlier detection to find if the equipment has faults. Outlier identification was developed based on mean and standard deviations. In this study, three categories of system fault were detected—chiller failure and improper control strategy, poor design of ventilating and air-conditioning equipment, and improper operation of equipment on the electrical panel. However, the proposed method could only be applied to certain system configurations, and could not easily be applied to other systems.

Jacob et al., [45] proposed another way to improve the outlier detection effectiveness. In their study, they used an improved linear regression models—change point multiple linear regression models to identify the abnormality in building daily energy consumption, and used the identified outliers to detect faults. Different from other research, they used indoor air temperature differences to reflect the energy variation and to improve the correlation coefficient of the model. Average daily energy consumption and peak energy consumption were chosen based on the previous tests.

Capozzoli et al., [50] proposed a) a statistical pattern recognition techniques to identify fault patterns, and b) ANN basic ensemble method and peak outlier detection method to detect outliers from building electricity consumption. Pattern comparisons for lighting power consumption and for total power consumption of a cluster of eight buildings were developed. Although the authors provided the basic structure of their developed ANN, such as the number of hidden layer and neurons, they did not provide a detailed description on the ANN model parameters. The detection accuracy was not reported in their article either.

Fuzzy sets is another effective way to describe the properties of objects which do not have clear boundary between good performance and bad performance. In the BAS, a large proportion of data points have a wide operational scale, which is more appropriate to be defined by fuzzy set. Yu et al., [42] proposed a method combining fuzzy set and neural network to detect faults by finding the overshoot of the threshold. But the selection of membership function is a challenging issue especially for the complex system.

1.4.5.4 Fault Diagnosis Methods

Fault diagnosis is the last and most challenging step in FDD. As stated above, for a whole building FDD, fault diagnosis is significantly more difficult because a detailed and accurate reasoning process has to be accomplished to find out the root cause. Currently, there are very few literature discussing whole building fault isolation.

In study [21], Wu et al., described several whole building faults (thermostat setpoint variation fault, VAV outlet blockage fault, AHU fan frozen, outside air temperature sensor fault, VAV supply air inlet blockage fault), but did not give a detailed description on the reasoning process.

Du [26] used energy model established by a neural network to find the fault sources respectively. These methods actually used subsystem abnormality comparison to identify the fault sources. The advantage of this method was that it provided a clear system classification and an intuitionistic analysis for field engineers. But the interrelations among the subsystem were not considered. Furthermore, this proposed diagnosis method will be limited when being used for a complex real system that is lack of subsystem information.

Magoulès [44] compared different data-driven methods and proposed a recursive deterministic perceptron neural network. In this method, different equipment faults were used as inputs to the energy model. Four error models were then trained to predict errors caused by four units including chiller, coil, fan and pump. When an equipment's error model demonstrated a good agreement with measured data, the associated equipment would be defined as the fault source. Again detailed information such as accuracy was not reported. The challenge of using such method is that it is very difficult to obtain fault data as training data.

Dodier et al., [24] used Bayesian belief network to diagnose the whole building energy usage fault. But Bayesian belief network depends on the basic knowledge of the system structure, and is limited when being used for a different building system.

1.5 Summary

As discussed above, component level FDD is more mature than whole building FDD. However, in real buildings, different subsystems often closely couple together and a fault that occurs in one subsystem may trigger abnormality in other equipment or systems. Hence, using component level FDD alone could result in high false alarm rate for those faults that have a whole building impact, due to the coupling among different subsystems and fault propagation. This is the reason why component level FDD is often less effective to isolate faults that affect multiple subsystems.

Furthermore, , most of the existing whole building FDD methods focuses on detecting whole building energy consumption abnormality, and lacks the ability to diagnose the root fault causes. More research are needed to develop cost-effective and scalable of diagnosis methods on whole building faults. Many of the reported FDD research are based on simulated data which could be

greatly different from how real systems behave. Therefore, more real building FDD testing and evaluation studies are desired to understand the performance of a FDD method.

Chapter 2 Demonstration Building Description

2.1 Demonstration Building –Nesbitt Hall Overview

One Drexel campus building – Nesbitt Hall is chosen as the demonstration building in this study. Nesbitt Hall is a seven-story, 78,000 square-foot mixed use building which includes offices, classrooms, laboratories, and an auditorium.

Figure 2-1 shows the appearance of the Nesbitt Hall. This building is chosen as the test building because its HVAC system has a configuration that is typical among medium- and large-size commercial buildings in the U.S. The HVAC system in the Nesbitt Hall includes a water cooled chiller system, three AHU systems that serve multiple VAV terminal units, and a hydronic heating system.

The primary cooling subsystem including one chiller and one variable speed pump, is located in the basement of the building. One cooling tower that serves the chiller is located in the penthouse. Three air distribution subsystems, which includes three AHUs (including one in the basement, other two on the rooftop) and eighty-eight VAV terminal units, are used to serve all seven floors. One steam-to-hot-water heat exchanger subsystem that is located in the basement is used to provide domestic hot water and space heating needs. Figure 2-2 illustrates the HVAC system configuration in the Nesbitt Hall. A BAS, which can remotely monitor and control the entire HVAC system, is connected to each subsystem.

Table 2-1 provides a summary of the subsystems. In the following sections, each subsystem is illustrated in details.

Figure 2-1 Overview of the Nesbitt Hall at Drexel University

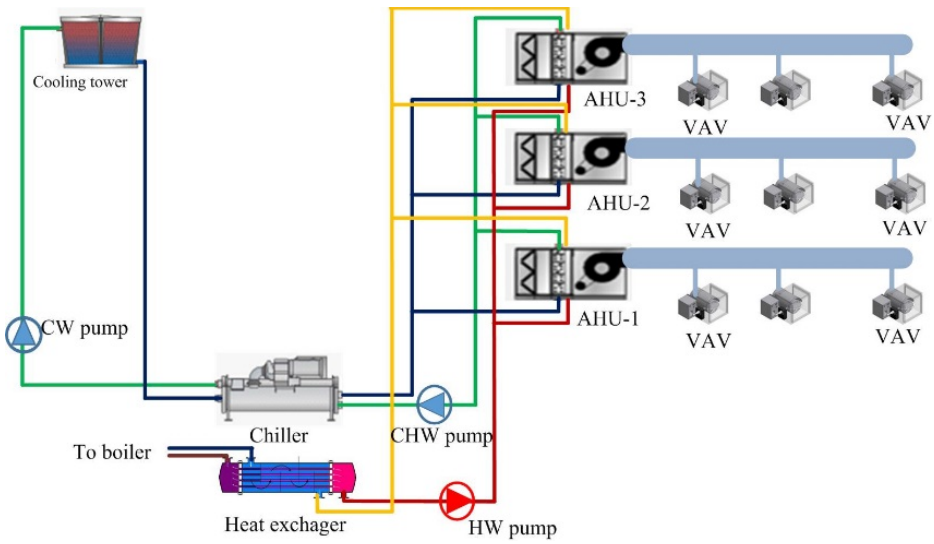


Figure 2-2 HVAC system configuration in the Nesbitt Hall

Table 2-1 BAS connected subsystem or measurement summary

Subsystem	Primary Cooling Subsystem	Supply air Subsystem	Steam-to-hot-water Heat Exchanger Subsystem	Whole Building Energy Measurement
Components	1 centrifugal chiller and 1 variable primary flow chilled water system	3 AHUs and 88 VAV terminal units	2 heat exchangers for domestic hot water and space heating water	Electric power meter

2.2 Overview of HVAC System

2.2.1 BAS in the Nesbitt Hall

The BAS for the Nesbitt Hall can be remotely accessed so that most of the control tasks can be implemented through remote computers. Data collected from various subsystems are stored in a cloud-base database and can be downloaded remotely.

Figure 2-3 demonstrates the BAS graphic interface. In the Nesbitt Hall, the BAS monitors the above-mentioned three subsystems. Furthermore, measurements from an electric power meter that measures the whole building electricity usage, as well as weather station sensors (temperature sensor and relative humidity sensor) are also connected to the BAS.

Outdoor air enthalpy information is not provided by the BAS, but is computed in this study by using the weather station temperature and relative humidity sensor measurements. Outdoor air pressure is also collected in this study through weather information website (https://www.accuweather.com/) during the outdoor enthalpy computation.

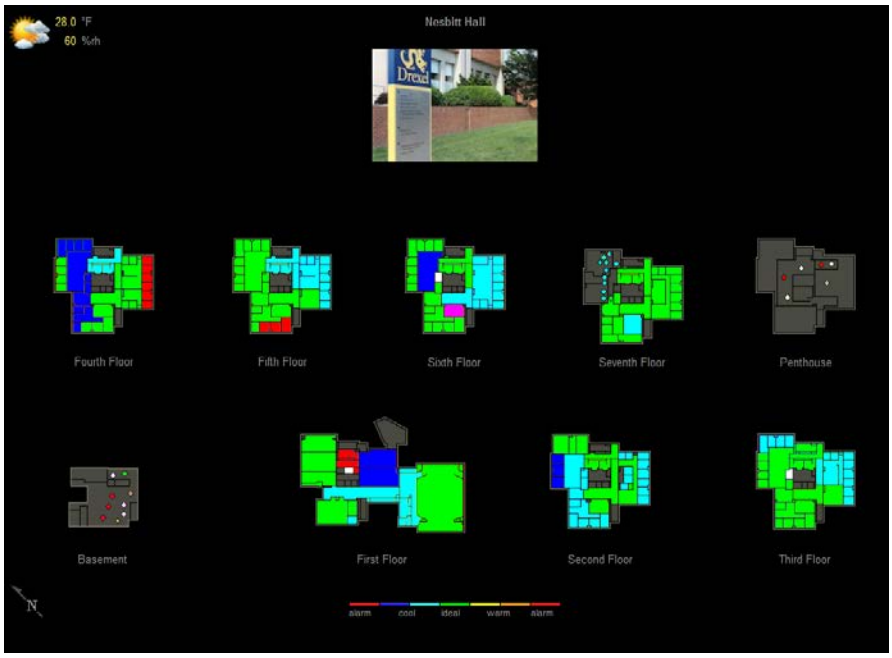


Figure 2-3 BAS graphic interface

2.2.2 Primary Cooling Subsystem

In the Nesbitt Hall, a Variable Primary Flow (VPF) system is employed in the chiller plant. The VPF system uses a variable speed pump (VSP) whose speed is controlled according to the differential pressure between chilled water (CHW) supply and CHW return pipes, so that the supplied chilled water flowrate can be adjusted according to cooling load variation. A CHW differential pressure sensor is used to measure the pressure difference. Figure 2-4 shows the VPF system configuration in the Nesbitt Hall.

In the chiller plant, one centrifugal chiller is employed to meet the cooling demand of the entire building as shown in Figure 2-5. One primary VSP and one spare pump (constant speed) are used to provide chilled water for three AHUs. A bypass control valve is used to ensure a minimum chilled water flowrate that the chiller requires to prevent excessive fouling and evaporator frozen when the chiller is operated under a low flowrate. Furthermore, a cooling tower is located on the penthouse to provide the cooling water (CW) as shown in Figure 2-6. Various sensors including water flowrate sensors, temperature sensors, pressure sensors, and pump speed sensors are installed in the chiller plant to collect data from each component.

2.2.3 Supply Air Subsystem

The air distribution subsystems include three AHUs and eighty-eight (88) VAV terminal units. Two of the AHUs (AHU-1 and AHU-2, Figure 2-7) are equipped with supply air duct and return air duct. AHU-3 (Figure 2-8) serves the laboratory space, so it does not include return duct. Dedicated exhaust air systems are installed in those labs. Figure 2-9 shows the appearance of AHU-1.

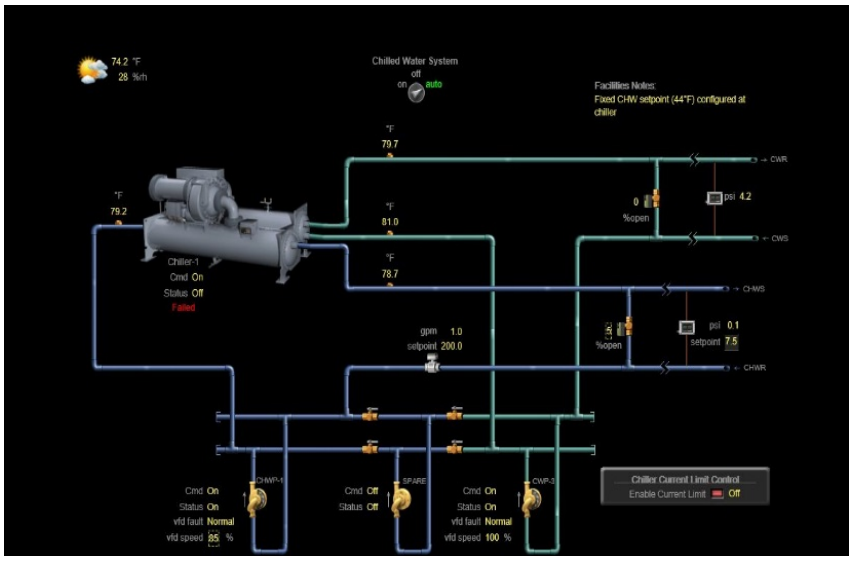


Figure 2-4 VPF primary cooling sub-system configuration in the Nesbitt Hall

Figure 2-5 Centrifugal chiller in the Nesbitt Hall

Eighty-eight (88) VAV terminal units are distributed in different zones as shown in Figure 2-10. The heating subsystem located in the basement of the Nesbitt Hall provides the hot water to the VAV reheat system.

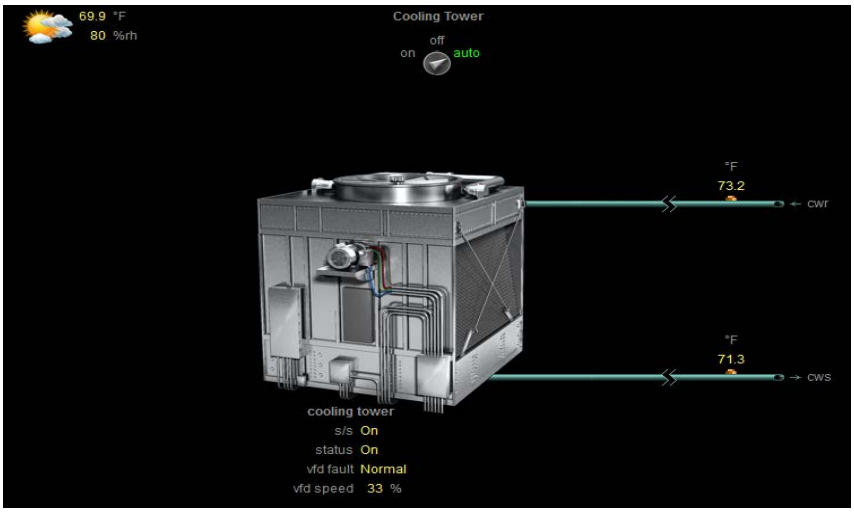


Figure 2-6 Cooling tower configuration in the Nesbitt Hall

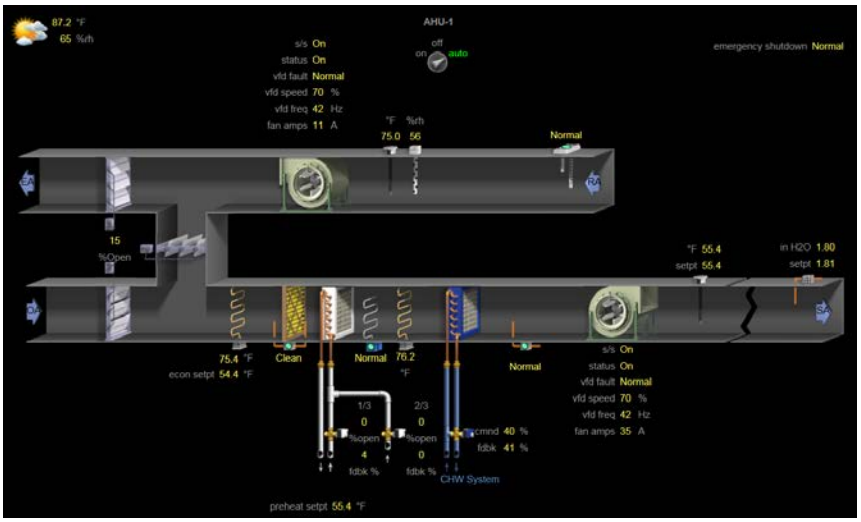


Figure 2-7 AHU with return air duct

2.2.4 Steam-to-hot-water Heat Exchanger Subsystem

Two steam-to-hot-water heat exchangers are included in the heating subsystem as shown in Figure 2-11. Two constant water pumps are equipped to provide the domestic hot water for each floor, and for each VAV terminal unit with a reheat coil. The BAS supervises the water temperature and pump operation. Water flowrate sensor is not included in the subsystem.

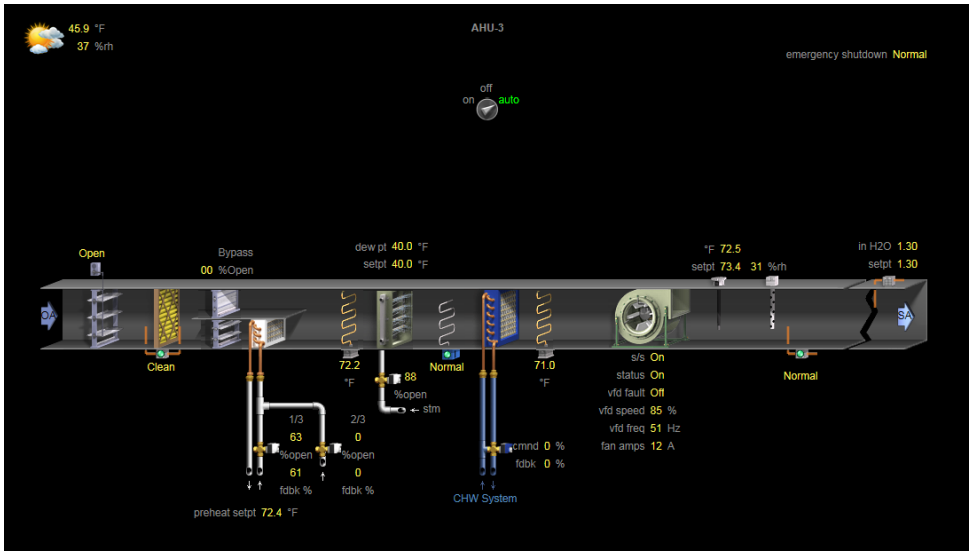
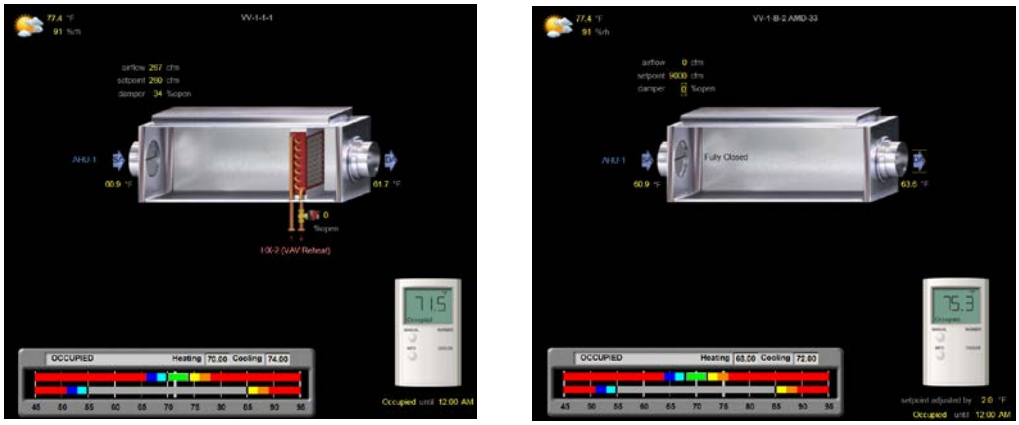


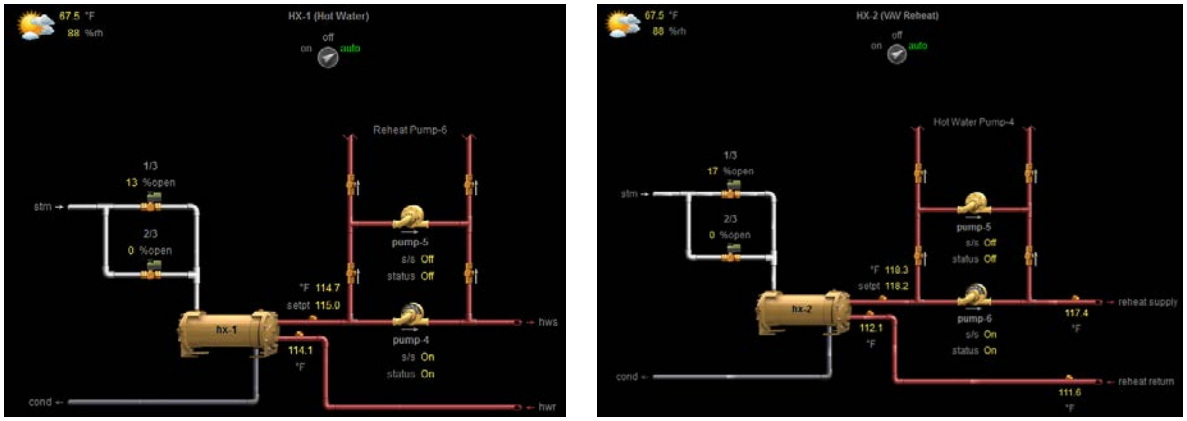
Figure 2-8 AHU without return air duct

Figure 2-9 AHU-1 profile



(a) (b)

Figure 2-10 VAV terminal unit (a) with reheat coil, (b) without reheat coil



(a) (b)

Figure 2-11 Heat exchangers (a) for domestic hot water (b) for VAV reheat

2.3 Summary

In this study, a demonstration building - Nesbitt Hall is identified as the test site. This building is equipped with typical HVAC systems that are commonly used in medium- and large-sized commercial buildings. The HVAC systems at Nesbitt Hall include a water cooled primary chiller subsystem, three VAV AHU subsystems, and a hydronic heating subsystem. A typical BAS is used to monitor and control the HVAC systems at Nesbitt Hall. The BAS is connected with a cloud-based server which allows remote control and data exchange. In next chapter, detailed test design and implementation processes are provided.

Chapter 3 Whole Building Fault Data Generation and Collection

3.1 Overview of Whole Building Fault Experiments

Based on the definition on whole building fault given in Chapter 1, a list of faults that have the potential to have whole building level impacts are identified firstly. However, not all of the identified faults can be implemented in the demonstration building due to safety considerations and facility capabilities. Only those whole building faults that can be implemented in the demonstration building and that do not have adverse indoor environmental impact are selected in the end as the focus of this study. These selected faults are categorized by the location of their root causes:

1) Primary cooling system faults;
2) Supply air system faults;
3) Software - Operator/scheduling faults.

Besides the faults that are implemented artificially, naturally occurred whole building faults have been observed in the data collected from the demonstration building. Table 3-1 provides a summary of the nine types of whole building faults, which are either manually implemented, or naturally occurred, in this study.

Table 3-1 Whole building faults implemented or natrually occurred in this research

Fault Location	Fault Name
Primary cooling system	Chilled water supply temperature sensor bias (screen reading higher/lower than real value)
	Chilled water loop differential pressure sensor bias (screen reading higher/lower than real value)
Supply air system	AHU outdoor air damper stuck at higher than normal position
	AHU supply air temperature sensor bias (screen reading higher/lower than real value)
	AHU cooling coil valve stuck at higher than normal position
Software - Operator/scheduling	System is unoccupied while under normal operation it should be occupied
	System is occupied while under normal operation it should be unoccupied
	Chiller is turned off while under normal operation it should be on
	AHU cooling coil valve software-overridden at a higher than normal position

Generally speaking, there are two types of methods to implement an artificial fault in a building system: a fault can either be implemented via the software, such as adding a sensor bias, or by changing the configuration of the hardware, such as using a voltage generator to output a constant voltage to a damper to keep its position unchanged, in an attempt to simulate a stuck fault. The later method produces operation data that are more similar to those from a naturally occurred fault. In this study, due to physical limitations, all faults are implemented via the BAS system, or through the chiller control panel. Both bias and frozen faults could occur to a sensor. Only bias

fault is considered in this study because of the above-mentioned implementation limitations at the Nesbitt Hall. There are two ways to implement a sensor frozen fault, i.e., either by disconnecting the sensor and feeding a constant signal to the control system (not doable at Nesbitt Hall); or by using software-override function to keep the sensor screen reading. If the later method is used, the fault symptoms will be the same as those software-override faults.

A total of thirty-four (34) fault tests have been implemented between Year 2016 and 2018. This includes twenty-seven (27) fault tests between Year 2016 and 2017 and seven (7) makeup tests in the summer of Year 2018 (discussed in Chapter 8). At the same, three naturally occurred fault cases are identified that have whole building impacts. Detailed description on all of the manually implemented fault tests, as well as the naturally occurred faults are provided in Appendix I. The following sections provide a detailed illustration of fault tests.

3.2 Primary Cooling System Fault

The chiller is controlled by a proprietary control system, and the BAS only monitors its performances. Therefore, the primary cooling system faults are implemented through the chiller's own control panel (chilled water temperature sensor bias fault) and the BAS (chilled water loop differential pressure sensor fault). Table 3-2 shows the fault test types implemented in the primary cooling system.

Table 3-2 Primary cooling system fault tests

Fault Name	Equipment	Operation Mode	Fault Direction
Chilled water supply temperature sensor bias fault	Sensor	Cooling mode	Negative bias
Chilled water loop differential pressure sensor fault	Sensor	Cooling mode	Positive bias

3.2.1 Chilled Water Supply (CHWS) Temperature Sensor Bias Fault

(1) Sensor function

CHWS temperature measurement is typically used to control a chiller's operation to achieve a desired CHWS setpoint. CHWS temperature setpoint can be reset according to outdoor environment information and indoor load.

(2) Implementation method

This fault is implemented by adjusting the CHWS temperature setpoint on the chiller control panel as shown in Figure 3-1.

(3) Expected symptoms

A CHWS temperature sensor bias fault will have an impact on the downstream AHU cooling coil valves' positions. If a negative bias is implemented, i.e., the screen reading is lower than the real value, the real CHWS temperature would be higher than its baseline values, which would

cause the downstream AHU cooling coil valves to have a higher than normal position. Depending on specific chiller's characteristics, such a fault could also have an impact on the chiller efficiency.

Figure 3-1 CHWS temperature setpoint configuration on chiller control panel

3.2.2 Chilled Water Differential Pressure Sensor Positive Bias Fault

(1) Sensor function

Chilled water loop differential pressure sensor is used to control the chilled water flowrate, which is modified by adjusting the variable frequency driver (VFD) chilled water pump to maintain a desired differential pressure setpoint.

(2) Implementation method

This fault test will be implemented through the control logic setting from the BAS system as shown in Figure 3-2.

Due to the difficulty of implementing a sensor's bias through the hardware or through editing the sensor parameters (the facility does not allow such modifications), a sensor bias fault is implemented through adjusting the associated setpoint. The chilled water differential pressure sensor bias fault test is implemented by adjusting the differential pressure setpoint value as shown in Figure 3-3. After setting a new setpoint value, the actuator will be operated according to the new setpoint value, as if a biased chilled water differential pressure measurement is taken. During the fault post-processing, the setpoint value is artificially adjusted to its original value. For example, if a differential pressure positive bias of 0.5 psi is implemented, the differential pressure setpoint is subtracted by 0.5 psi. During the post-processing procedure, a value of 0.5 psi should be added to the trended setpoint value.

(3) Expected symptoms

A differential pressure sensor bias fault affects the VFD pump speed and has an impact on the pump's energy consumption. At the same time, AHU cooling coil valve's opening position will be also influenced by this fault as the chilled water supply pressure is different from its baseline

value. A differential pressure sensor positive bias fault may increase the AHU cooling coil valve opening position to be at a higher than normal position.

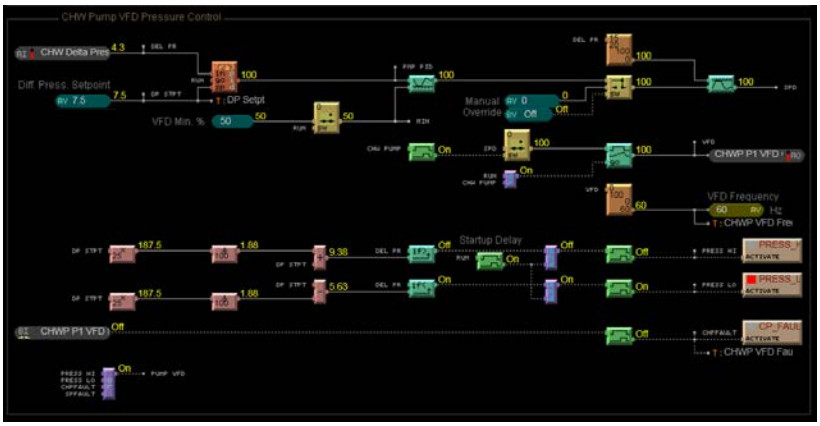


Figure 3-2 Chilled water supply differential pressure sensor logic configuration interface

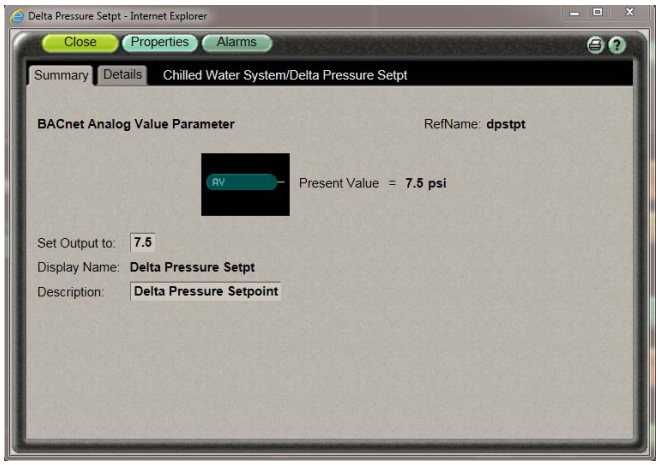


Figure 3-3 Chilled water differential pressure setpoint configuration

3.3 Supply Air System Fault

Faults in a supply air system could will affect both the AHU and VAV terminal units and might affect the primary cooling system too. In this study, three supply air system faults that could have a whole building impact are identified and implemented in the demonstration building. Table 3-3 lists the three faults.

3.3.1 AHU Supply Air Temperature Sensor Bias Fault

(1) Sensor function

AHU supply air temperature sensor is typically used to control the AHU components to maintain a desired supply air temperature setpoint.

(2) Implementation method

This fault is implemented through the BAS by adjusting the demand adjust (DEM ADJ) to change the supply air temperature setpoint as shown in Figure 3-4. For example, if a negative bias of 2 °F fault is implemented, the value of DEM ADJ is increased by 1. Later when preprocessing the collected data, a value of 2 should be subtracted from the supply air temperature setpoint and supply air temperature readings. In this way, the screen reading value is lower than the real supply air temperature value.

Table 3-3 Supply air fault tests

Fault Name	Equipment	Operation Mode	Fault Direction
AHU supply air temperature sensor bias	Sensor	Cooling	Sensor screen reading is higher/lower than real value
AHU outdoor air damper stuck at higher than normal position	Actuator	All modes	Outdoor air damper is stuck at a position that is higher than its normal position under similar weather/internal load conditions
AHU cooling coil valve stuck at a higher than normal position	Actuator	Cooling	Cooling coil valve is stuck at a position that is higher than its normal position under similar weather/internal load conditions

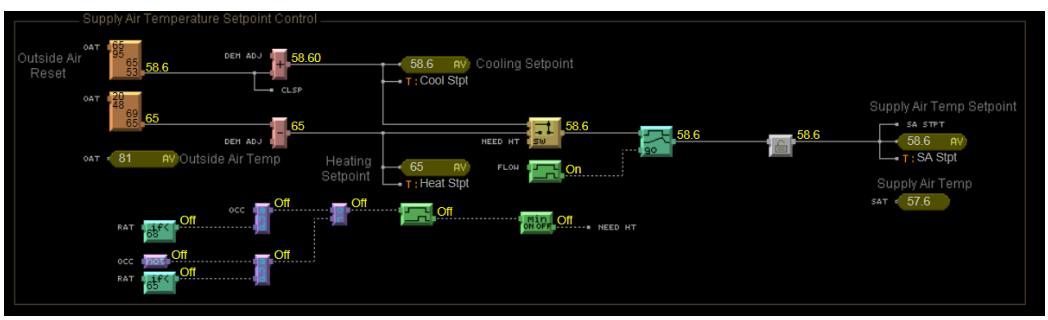


Figure 3-4 AHU supply temperature setpoint configuration

(3) Expected symptoms

When there is an AHU supply temperature sensor bias fault, the AHU cooling or heating coil valve positions and fan speed will be different from their baseline values. For example, in a cooling mode, the cooling coil valve position will be lower than its baseline value under similar load and operation conditions, if there is a supply air temperature sensor negative bias fault. At the same time, the supply air fan speed could be higher than the baseline value to provide more air flow needed by the downstream zones.

3.3.2 AHU Outdoor Air Damper Stuck at a Higher than Normal Position Fault

(1) Damper function

The AHU outdoor air damper is firstly controlled to achieve a desired ventilation rate. In the heating and/or cooling (non-economizer) mode, the outdoor air damper is typically maintained at a minimum opening position (e.g. 15% open at the demonstration building) to satisfy the ventilation requirement. In an economizer mode, such as when outdoor air temperature is lower than the return air temperature, and outdoor enthalpy is lower than the return air enthalpy (which are the conditions used at the demonstration building), the outdoor air damper is either at 100% open (if the outdoor air temperature is higher than the supply air setpoint) or adjusted to maintain the mixed air or supply air setpoint.

(2) Implementation method

This fault is implemented through the BAS by locking the software point “Economizer” to a desired value as shown in Figure 3-5.

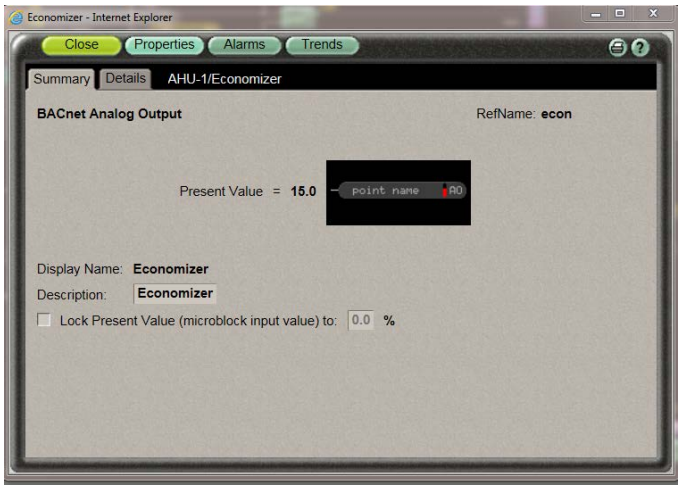


Figure 3-5 AHU economizer adjustment

(3) Expected symptoms

When the outdoor air damper is stuck, the heating/cooling loads brought in by the outdoor air would be different from their baseline values. Hence the AHU heating/cooling coil valve positions would be abnormal. If the damper stuck position is very different from the normal position, for example, the damper is stuck at 100% open (rather than the minimum position) in a hot summer day, the load could be abnormal enough to be reflected in the primary cooling system operation, resulting in energy and chilled water measurement abnormality. The supply air temperature setpoint may not be maintained successfully, which would result in downstream zone temperature abnormality.

3.3.3 AHU Cooling Coil Valve Stuck at a Higher than Normal Position Fault

(1) Cooling coil valve function

An AHU's cooling coil valve is controlled according to the supply air temperature setpoint to maintain a desired supply air temperature. A motorized driver is typically used to automatically adjust the valve position based on a voltage signal provided by a controller.

(2) Implementation method

As discussed earlier, this fault cannot be implemented by adjusting the voltage signal to the driver due to the limitations in the demonstration building. This fault is implemented by overriding the cooling coil valve signal through BAS as shown in Figure 3-6. Hence the implementation of this fault is the same as that from an operator fault – software override AHU cooling coil valve position (Section 3.4.1). When this fault occurs naturally, the symptoms of this fault are very similar to the operator fault too, except one difference: when the valve is stuck naturally, the screen reading of the valve would be the same as the controller output, although the real position is stuck. However, for a software override fault, the screen reading of the valve position is the overridden value, not the controller output. Hence, when this fault is implemented, the controller output is manually recorded and used instead of the trended valve position.

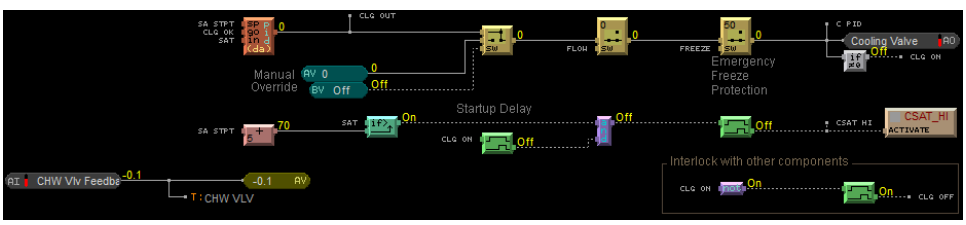


Figure 3-6 AHU cooling coil valve adjustment

(3) Expected symptoms

If the AHU cooling coil valve is stuck at a higher than normal position, it will cause the supply air temperature to be lower than its setpoint, and therefore introduce more cooling energy consumption. To the upstream, a higher openness of the cooling coil valve causes the speed of the primary chilled water VFP to be increased, and the chilled water flowrate is increased as a consequence. At the same time, the cooling coil valve controller output signal may be abnormal compared with the baseline attempting to drive the valve to the correct position.

3.4 Operator Fault

Operator fault is to simulate the faults from the operators or field engineers during the course of system commissioning or maintenance. In the real practice, operator fault mainly includes two categories. The first category is that the operation schedule is wrongly set by field engineers. The second category is that the faulty setpoint/control signal in HVAC system is wrongly configured. Table 3-4 lists the operator fault tests that will be implemented.

Table 3-4 Operator fault test

Fault Name	Equipment	Operation Mode	Fault Direction
AHU software wrong: cooling coil valve to be a higher than normal position	System	Cooling	Override software setting
System scheduling fault	System	All modes	Change system schedule

3.4.1 AHU Cooling Coil Valve Software Operator Override Fault

(1) Cooling coil valve function

The AHU cooling coil valve is controlled according to the supply air temperature setpoint to provide required supply air temperature. The valve position is controlled by the valve driver.

(2) Implementation method

The implementation of this fault is same as cooling coil valve stuck at a higher than normal position fault.

(3) Expected symptoms

If the AHU cooling coil valve is overridden at a higher than normal position, it causes the supply air temperature to be lower than its setpoint, and therefore introduces more cooling energy consumption.

3.4.2 System Scheduling Fault

A HVAC system is operated based on its occupancy schedule settings. Therefore, an error in the occupancy schedule setting will have a significant energy impact and lead to whole building abnormality.

This fault test is implemented through BAS system as shown in Figure 3-7.

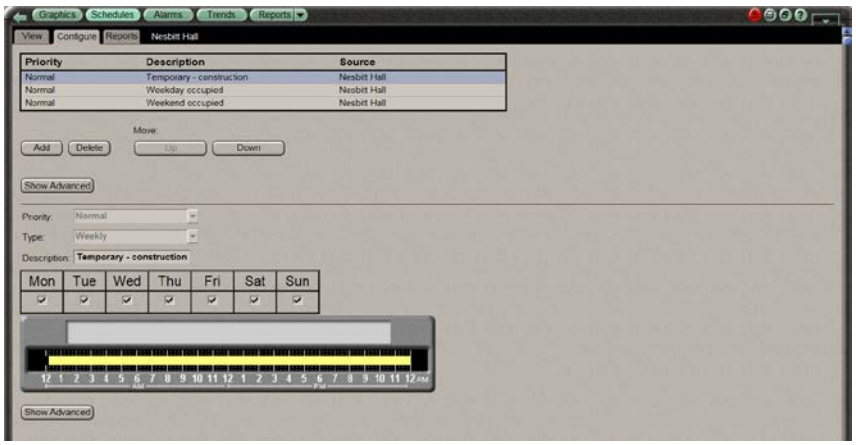


Figure 3-7 System scheduling configuration

3.5 Fault Implementation Consideration

When implementing a fault, special considerations are given to avoid adverse impacts on the system or indoor environment quality. A fault test day is selected after carefully examining the weather conditions. At the same time, the zone conditions and targeted equipment on which the fault is implemented are also carefully monitored during the test to ensure the safety of occupants and equipment.

Observations of system performance as well as zone temperatures are recorded manually on an hourly basis and a test will be terminated if any of the following conditions are met:

- Any zone temperature in the test areas exceeds 80°F ;
- Any zone temperature in the test areas drops below 60°F;
- The outdoor air damper of an AHU is fully closed;
- The supply air fan and return air fan are shut down abnormally;
- Any alarms in the system is triggered.

3.6 Fault Implementation Methods

A daily test procedure as shown in Figure 3-8 is followed for each test case.

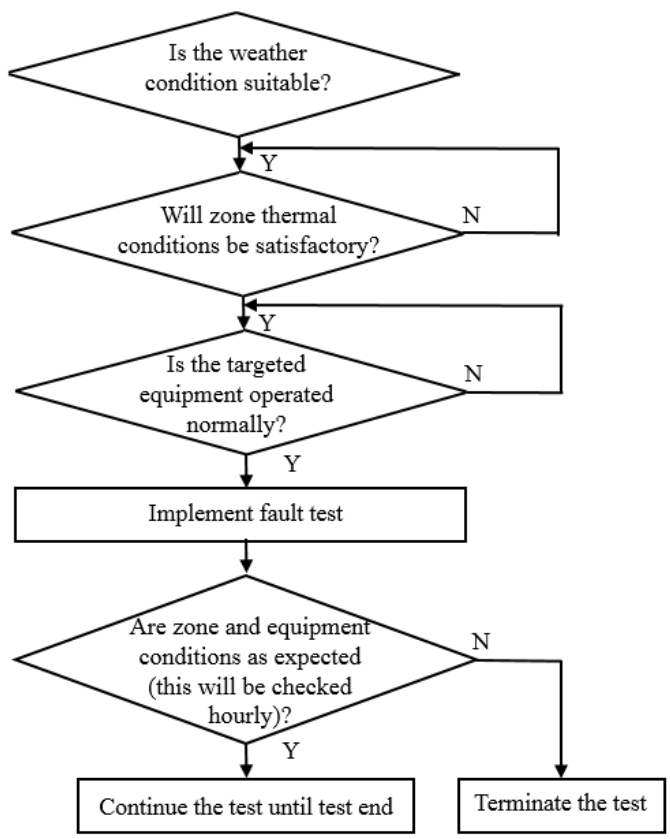


Figure 3-8 Fault test implementation flowchart

Weather conditions are the first consideration in the checklist to ensure that the fault test will not be implemented in any extreme weather conditions. The next step is to check the zone thermal conditions to ensure that the implemented fault will not cause any complaints from the occupants. Usually, from the zone condition check, equipment operation information can be evaluated as well. If zone thermal conditions are not maintained well, a naturally occurred fault might already exist. In the third step, equipment conditions will be examined, again, to identify if a naturally occurred fault exists. When all conditions are satisfied, the fault test begins and the test observations are manually logged during the course of the test.

3.7 Summary

In this Chapter, a variety of whole building faults, including primary cooling system, supply air system and operator faults, are discussed in details. Considerations that need to be given when implementing these whole building faults are explained. Fault implementation procedure is also illustrated in details.

Chapter 4 Development of Fault Detection Method

4.1 Introduction

In a building AFDD process, the process of fault detection identifies whether the building system is abnormal when compared with a pre-defined baseline. As discussed in Chapter 1, there is a need for data-driven fault detection strategies that can overcome the difficulties associated with physical model based methods such as high engineering cost and low scalability.

Although a variety of data-driven methods for component level fault detection has been successfully developed in the past two decades, it is difficult to directly implement these methods for whole building fault detection because of the following challenges: 1) complicated subsystem coupling; 2) much larger dataset dimension; and 3) difficulties to differentiate abnormalities that are triggered by weather/occupancy from those triggered by faults.

Targeting these three challenges, this research proposes a novel multi-structure fault detection method which consists of 1) a whole building feature selection process that aims at identifying key features to reflect system coupling and to reduce dataset dimension, 2) a Weather and schedule information based Pattern Matching (WPM) process that aims at differentiating the abnormality caused by weather/occupancy from those by faults; and 3) a Feature based Principal Component Analysis (FPCA) fault detection process that aims at achieving high automation, cost-effectiveness, and scalability.

The goal of this new Weather and schedule information based Pattern Matching and Feature based Principal Component Analysis (WPM-FPCA) fault detection method is to monitor whether a building's performance is significantly different (abnormal) from a pre-defined baseline (normal). Since it is a data-driven fault detection, "fault free" data that represent the normal condition need to be collected firstly. As discussed in Chapter 1, it is understood that there is no true "fault free" status for a real building, here the condition of "fault free" is defined as a satisfied condition, such as when the building is just commissioned.

4.2 Overview of the WPM-FPCA Fault Detection Method

More specifically, the feature selection process in the developed whole building WPM-FPCA fault detection method uses Partial Least Square Regression (PLSR) and Genetic Algorithm (GA) method. Symbolic Aggregate Approximation (SAX), which is a highly efficient PM method designed for time-series data, is adopted for the WPM. Figure 4-1 illustrates the overall architecture of the developed strategy.

Firstly, historical baseline operation data as described earlier are collected. A PLSR and GA based feature selection process is implemented by using the collected historical baseline data. Through this process, the dimension of the original building data set is reduced and only useful measurements are remained for the later fault detection.

Secondly, a WPM method is developed to search the historical baseline data and generate a WPM baseline dataset, which has similar weather and building use (operation and occupancy) conditions as the incoming snapshot data (current data collected from the building). In a building, its HVAC system is operated in different operation modes under different working conditions such as weather conditions and occupant conditions. Hence, the historical baseline data may vary under

different operation modes. This poses a challenge for the data-driven method as it is difficult to develop dynamic baseline model. In this study, weather and schedule information are used to during a pattern matching process to identify the system operation data under the same weather and schedule patterns, which are then grouped to generate a WPM baseline dataset for each snapshot period.

The third process is to detect whether a whole building level abnormality exists. PCA is employed here to generate statistic thresholds from the WPM baseline dataset and compare with those from the incoming snapshot data. When the statistical indices surpass the thresholds, the building is flagged as faulty.

In the following sections, these three processes are illustrated in details.

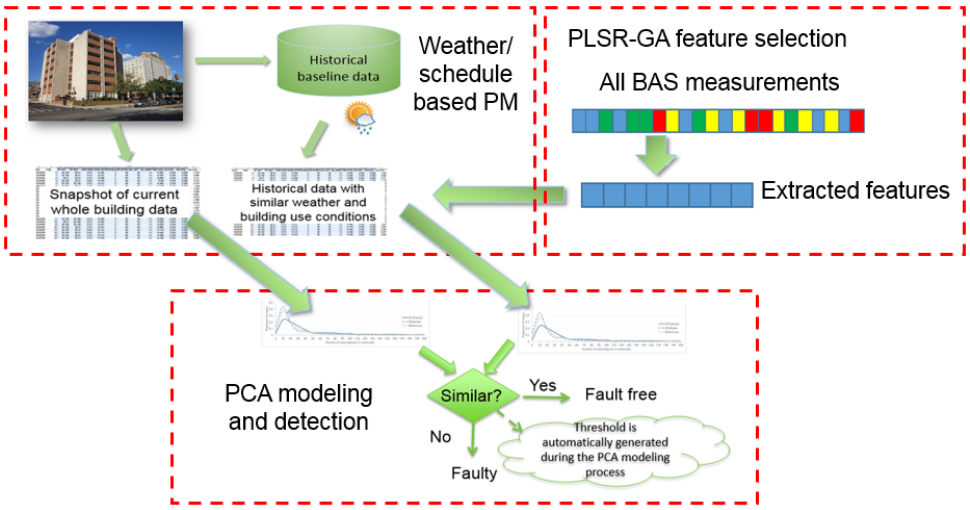


Figure 4-1 Architecture of WPM-FPCA fault detection method

4.3 Development of Feature Selection Process

4.3.1 Introduction

4.3.1.1 Purpose of Variable Dimensionality Reduction

Although, large quantity of system measurements can be collected and stored in a BAS, not all the measurements are equally important for monitoring the system. Usually, useful information are included in few key measurements for monitoring the building system operation or detecting system faults. This leads to a “data rich, but information poor” situation in current building systems [51]. Furthermore, a large number of data measurements in a dataset may lower the accuracy of a fault detection method and increase the computational burden [52, 53]. Therefore, it is necessary to identify the key variables (features) which contain more useful information than other variables for building system status monitoring, before developing a fault detection method [54].

In practice, some key variables can be defined as features of the system. Therefore, the determination of these key variables turns out to be a process of reducing variable dimension. This

process can be achieved by optimally designing the monitoring system using expert knowledge, i.e., an expert determines and pre-selects a specific subset of variable dataset for monitoring system operation [54]. However, in a large scale system, manually selecting the proper variable subset group and reducing the variable dimensionality become challenging for the following two reasons. Firstly, the number of data measurements has been increased rapidly with the increasing complexity of the system. A manual identification process will take too much time. Secondly, the system with multi-operation modes generates various combination of sub dataset. This makes the expert knowledge based variables reduction impractical in the real practice. Moreover, for building systems, key variables often vary from building to building due to the diversity of building design, envelope types, internal load, and system configurations, which makes manual variable reduction process much more difficult and inaccurate. Hence, automated variable dimension reduction techniques are the focus of this study. Existing variable dimension reduction techniques can be divided into two categories as feature selection and feature extraction.

The purpose of feature selection is to reduce the dimensionality of data without losing intrinsic information. Usually, high dimension data is mapped to a lower dimensional space. By this means, the useful information is retained and uninformative variance in the data is discarded [55]. At the same time, the computation efficiency can be increased as less variables are included in a building model or a monitoring system.

Usually, the process of feature selection is to choose a subset of M features from the original set of N features ($M \leq N$), so that the feature space is optimally reduced to meet a certain criteria [56]. When implementing feature selection method, two aspects needed to be considered. One aspect is to develop an efficient search strategy to select candidate subsets. Another aspect is to establish an objective function to evaluate the candidate subsets [57]. Different evaluation criteria have been used to assess the optimal candidate subsets. For example, in a supervised learning, one of the primary goals of feature selection is to use the optimal feature subsets to maximize predictive accuracy [58]. Moreover, some performance measurements such as similarity factor, extraction accuracy can be employed to fulfill the goal of feature extraction when searching for a minimum set of new features [59].

4.3.1.2 Category for Feature Selection Method

Various feature selection methods have been developed to reduce the data dimension. A common taxonomy is to categorize the feature selection techniques into three types as “filter”, “wrapper” and “embedded” as shown in Figure 4-2 [60]. This kind of taxonomy was also used by Mehmood et al., [61] when dealing with variable selection methods used for PLSR in the process control.

Here, “*filter*” methods identify the relevance of variable by only assessing the intrinsic properties of the data. In this kind of method, subsets of variables are selected as a preprocessing step and this step is independent from the chosen predictor. A “*wrapper*” method incorporates the model hypothesis search within the feature subset search. Therefore, a learning mechanism is included to score the variable subsets according with the modeling target. The “*embedded*” methods include the variable subset searching method in the classifier construction. So, embedded methods are specific to a given learning algorithm.

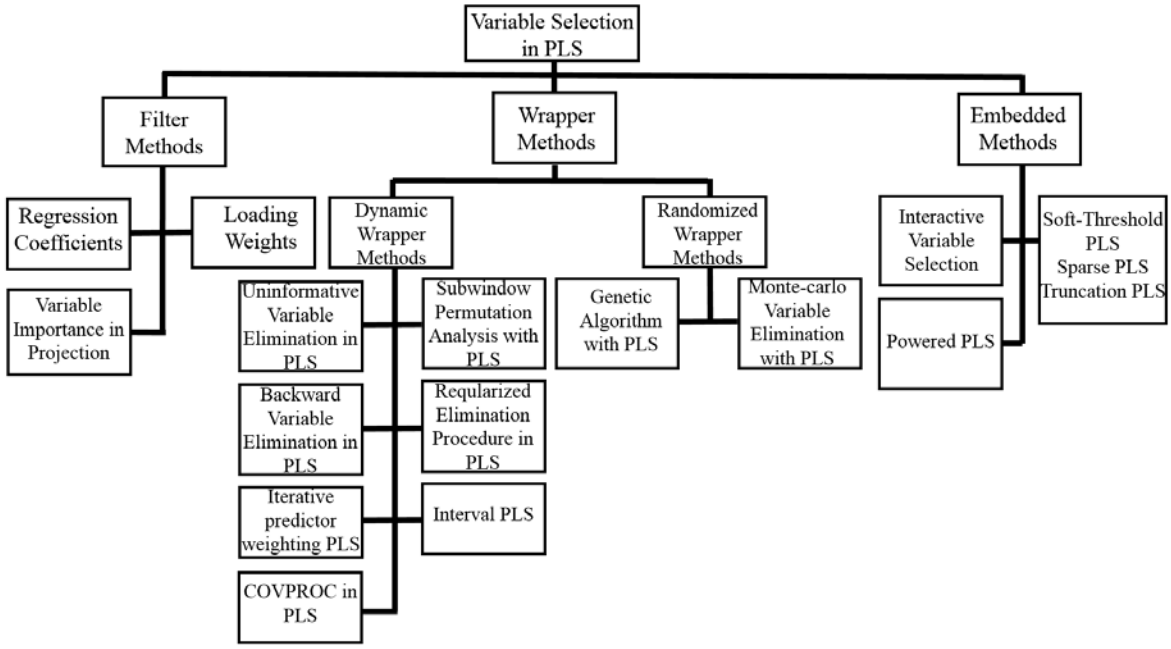


Figure 4-2 Categories of dimension reduction method (adapted from [61])

4.3.1.3 Basic Feature Selection Flow

Although, a large number of dimension reduction methods have been developed in the past three decades, very few studies on a unified work flow have been found in the existing literature. This is probably because dimension reduction methods are used to solve problems in different practical fields, so it would be very hard to find a unified workflow to employ them. However, one general step checklist was proposed when implementing variable selection [62].

4.3.2 Partial Least Square Regression (PLSR)

4.3.2.1 Introduction of PLSR

In this research, a feature selection method is used to identify candidate key variables which represent the system performance. This method employs a PLSR technique to develop system performance model. Building energy consumption from fault free days is used as the target output variable (predicted variable). System operation variables are used as predictors. GA is used to search whole variable set and generate candidate variable subset for the PLSR. When the system performance standard is met, the variable subset is determined. Through this process, the key informative variables can be selected and the variable dimensionality can be reduced.

PLSR is a technique that generalizes and combines features from PCA and multiple regression. Although, this technique was not originally designed for classifying and discriminating the dataset, it was found to be efficient to distinguish different datasets and determine the key variables [63].

Suppose system output **Y** matrix ($Y \in R^{M\times 1}$) consists of M observations and one dependent variable, and system input **X** ($X \in R^{M\times \mathrm{N}}$) matrix consists of M observations and N dependent variables. The goal of a PLSR is to best predict **Y** from **X** and to describe their common structure [64]. Usually, PLSR searches for a set of components (called latent vectors) that decompose

simultaneously the **X** and **Y** with the constraint that these components maximally explain the covariance between **X** and **Y**. This step is similar as in the principal component analysis. By this means, the system input **X** matrix can be decomposed as:

$$\boldsymbol{X} = TP^T + E \text{ with } T^T T = I \tag{1}$$

where $T \in R^{M\times \mathrm{K}}$ is the score matrix, $P \in R^{M\times \mathrm{N}}$ is the loading matrix, $\boldsymbol{E} \in R^{\mathrm{M}\times \mathrm{N}}$ is the error matrix and $\boldsymbol{I}$ is the identity matrix.

The output can be decomposed as:

$$\boldsymbol{y} = Tb + f \tag{2}$$

where $\boldsymbol{b} = [b_1, \cdots, b_k]^T \in R^{\mathrm{K}}$ is the loading vector of $\boldsymbol{y}$, $\boldsymbol{f} \in R^{\mathrm{M}}$ is the error vector.

A nonlinear iterative partial least squares algorithm is employed to develop the Partial Least Square (PLS) model [65]. Suppose that the first latent $\boldsymbol{t_1}$ variable to the *k-1*th latent variables t_{k-1} , the loading vectors $p_1 \ldots p_{k-1}$and the loading vector $b_1 \ldots b_{k-1}$are given. Then the *k*th residual input and output can be written as follows:

$$X_k = X_{k-1} + t_k p_{k-1}^T \tag{3}$$

$$y_k = y_{k-1} + b_k t_{k-1} \tag{4}$$

where t_k is a linear combination of the columns of X_k. $t_k = X_k w_k$, and w_k is the *k*th weighting vector. It is defined so that the covariance between y_k and t_k is maximized under $||w_k|| = 1$.

By using the Lagrange multipliers method, the function to maximize y_k and t_k can be defined as [66]:

$$G_k = y_k^T t_k - \mu(\,||w_k||-1) = y_k^T X_k w_k - \mu(\,||w_k||-1) \tag{5}$$

where μ is the Lagrange multiplier. Solving $\frac{\partial G_k}{\partial \mathrm{w}} = 0$, w_k can be derived as

$$w_k = \frac{X_k^T y_k}{||X_k^T y_k||} \tag{6}$$

The *k*th loading vector p_k and the k loading b_k can be written as:

$$p_k = \frac{X_k^T t_k}{t_k^T t_k} \tag{7}$$

$$b_k = \frac{y_k^T t_k}{t_k^T b_k} \tag{8}$$

In the PLSR, the entire procedure is repeated until the number of adopted latent variables *K* is obtained. *K* can be determined by the cross-validation.

4.3.3.2 Wrapper Method in PLSR for Feature Selection

There are many PLSR algorithms for feature selection. Most methods are based on orthogonal score of the PLSR. Wrapper method [67] is one of the most popular methods among them.

Wrapper method is implemented in an iterative way, and is based on certain supervised learning approach, where model refitting is wrapped within the variable search algorithm. Variable searching algorithm is used to obtain the subset of candidate variables and evaluate each subset by fitting a system performance model to the variable subset. Figure 4-3 illustrates the procedure of a wrapper method. Based on the different searching strategies for selecting variable group are used, wrapper methods can be categorized into deterministic types or random type [60]. In a randomized searching algorithm, some kind of randomness strategies such as PLSR-GA [68] and the Monte-Carlo based uninformative variable election and PLS [69] are used to select variable subsets. In a deterministic search based wrapper method, the variable subset is chosen by a deterministic mechanism including backward variable elimination PLS [70], sub-window permutation analysis coupled with PLS [71] and so on. Although deterministic search based wrapper methods are relatively simpler and require less computations, they are more prone than randomized searching algorithms [60]. In this research, a GA based randomized searching method is employed in PLSR for feature selection.

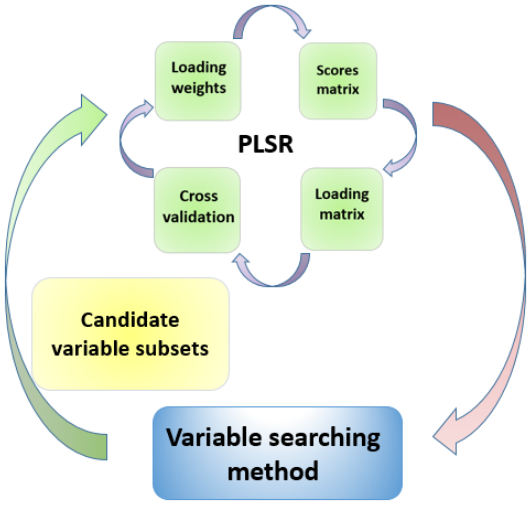


Figure 4-3 Illustration of wrapper method for feature selection

4.3.3 GA Searching

As discussed above, when the data variable set is very large, there is a need to use variable subset searching algorithm to improve the variable selection efficiency. Here, a GA search is employed to facilitate the process of searching candidate variable subsets. GA searching acts in an iterative way to extracts the subset of relevant variables and evaluate each subset by fitting a model to the subset variable.

GA has an advantage to solve multi-objective optimization problem when a set of solutions are found to be superior to the rest of the solutions in the search space [72]. Through the process similar to natural selection, high-quality solutions to optimization and search problems could be generated through the operators such as mutation, crossover and selection defined in field of biology [73].

In a GA searching algorithm, a given variable subset is represented as a binary string of length *n*. Here n is the total number of the available variables and the string can be mimicked as "chromosome" in biology [59]. In position index *i* of the string, a zero or one will denote the absence or presence of variable *i* in the variable set. Through this way, a population of variable subsets (chromosomes) is maintained. The "fitness" is used to determine how likely the variable set can survive and breed into the next generation [74]. New variable set are created from old variable sets by the processes of similarly with cross-over and mutation. In the process of cross-over, two different parent variable set are mixed together to create an offspring variable subset. In the process of mutation, the indices of single of parent variable set are randomly selected to create an offspring variable subset. The iteration will continue through these two process until the ending criteria are met. This criteria can be either a finite number of iterations or a certain percentage of the individuals in the population using identical variable subsets.

4.3.4 PLSR-GA based Feature Selection

4.3.4.1 Flow of PLSR-GA based Feature Selection

The procedure of using GA searching algorithm in PLSR for feature selection usually includes the following steps:

- Generate random variable subsets. In this step, initial population of variable sets is established by setting bits for each variable randomly, where bit '1' represents selection of corresponding variable while '0' presents non-selection. The approximate size of the variable sets is defined in advance.
- Evaluate each individual subset of the selected variables. In this step, a PLSR-model is fit to each variable set and the model performance is computed.
- Discard worse half of individuals. After evaluate the variable subsets, the worse half of variables are eliminated.
- Using cross-validation to evaluate the model performance.
- A collection of variable sets with higher performance are selected to survive until the next "generation".
- Crossover and mutation: new variable sets are formed 1) by crossover of selected variables between the surviving variable sets, and 2) by changing (mutating) the bit value for each variable by small probability [61, 75].
- Selected variable subset is determined.

Figure 4-4 illustrates the flowchart of PLSR-GA based Feature Selection.

4.3.4.2 Cross Validation

When implementing GA variable selection, a cross-validation is performed for each of the variable subset to determine which subset will be remained. In this study, contiguous blocks method is used in cross validation because for time-series data and batch data, this method can be convenient for assessing the temporal stability and batch-to-batch stability of a model built from the data. In contiguous blocks method, each test sets is determined by selecting contiguous blocks of n/s objects in the data set [76].

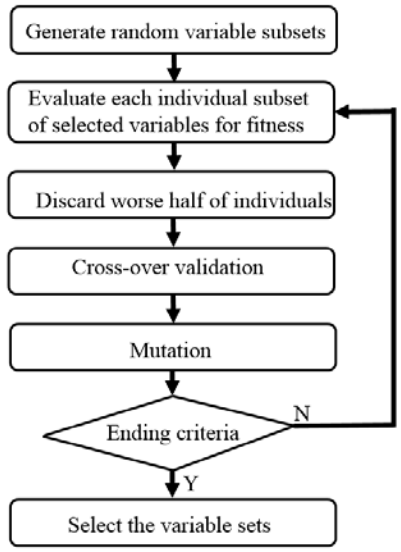


Figure 4-4 Flow of GA for feature selection

The variable subset which provides the lowest model root-mean-square error of cross-validation (RMSECV) is selected [77]. RMSECV is a "composite" prediction error obtained over all repetitions of the cross-validation procedure as given below.

$$RMSECV = \sqrt{\frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{n}} \tag{9}$$

where $\hat{y}$ is the values of the Y variables that are estimated by cross validation. n is the total number of objects in the date set.

4.3.4.3 Selection of GA Parameters

Two problems exist when using GA method to select variables. The first is model over-fitting which means the variable subset selected may be good for the model generated from the given data, but less accuracy for the unknown data. In order to solve this problem, random validation or multiple iterations are usually used when using GA method to select variable. The second problem is that GA method is usually time consuming. For example, the computation burden will be increased when a large population size is used in each generation. So, the GA parameters should be decided before employing GA method. So it is very important to choose a set of proper parameters used for GA method. The following key parameters were tuned when using GA method [77].

• **Size of Population**: Population size represents the different variable combination. Although, a larger population provides a better representation of different variable combinations, more computation time will be required during each iteration.

• **Window Width**: When adjacent variables contain correlated information, or the end use of the variable selection is to include or exclude adjacent variables together as a block (*e.g.*, lower-resolution measurement), the original variables can be grouped together and included or excluded

in "blocks". Window width indicates how many adjacent variables should be grouped together at a time. The grouping starts with the first variable and the subsequent (window width-1) variables.

• **Initial Terms**: This specifies the approximate number of variables (terms) included in the initial variable subsets. Starting with fewer initial terms will make the identification of useful variables more difficult, but will bias the end solution towards models with fewer variables.

Final parameters used in variable selection tests are listed in Table 4-1.

Table 4-1 PLSR-GA parameter setting

PLS-GA Parameters Setting	Value
Size of Population	64
Window Width	1
Initial Terms	30
Max Generations	100
Percent at Convergence	50
Mutation Rate	0.005
Regression Choice	PLS
Number of Latent Variables	10
Cross Validation Setting	Contiguous blocks

4.3.5 Variable Selection Result

It is noted that for a building system, it is quite difficult to choose one or two measured variables as the benchmarks to assess the overall system performance of a building. But, as state above, it is probably also unpersuasive to choose a similar factor which measures two datasets to evaluate a building system performance. So, it may be reliable to choose a few of key variables to describe the system performance. Other predictive variables can be indoor air quality, chiller COP and etc. But unfortunately, these value cannot be obtained at hand.

In the variable selection process, whole building electricity consumption is used as predicted variable. 537 data measurements collected from BAS in the Nesbitt Hall are used as predictor. Data collected in the Nesbitt Hall in winter season and summer season are used to obtain the selected feature. Operating data in every 5 fault free day from each season is used to in one feature selection scenario. As the data sample rate in the Nesbitt Hall is 5-minute, this accounts for 288 data samples in one operation day. Therefore, in one selection case, 1440 data samples are used to develop a PLSR model. In the contiguous blocks cross validation method, data sample set is split into *s* blocks (we use *s* equals 4). Therefore, in each cross validation, 360 data samples are used to build the model, and the 360 data samples are used to test model.

In each season, 12 scenarios (5 days in each scenario) are used to evaluate the feature selection result. Figure 4-5 shows a variable selection result from one scenarios. Operation data from five fault free from August, 2016 in the Nesbitt Hall are collected for PLSR-GA feature selection. In figure (a), it shows the number of selected variables at the end of generation. Figure (b) shows model evolution trend accord to the generation of each variable group. The read dash line shows the model fitness (with the lowest RMSECV) by using all variables. The read solid line shows the predicted best fitness. The blue line shows the actual fitness at each step of the generation. Figure

(c) shows the number of variable used in each generation. Figure (d) show the contribution of each variable to the predicted model. The larger value the variable used in the model, the higher possible that this variable will be selected. From this Figure, it can be seen that the model fitness increases when fewer variables are selected, and the data dimensionality can be reduced.

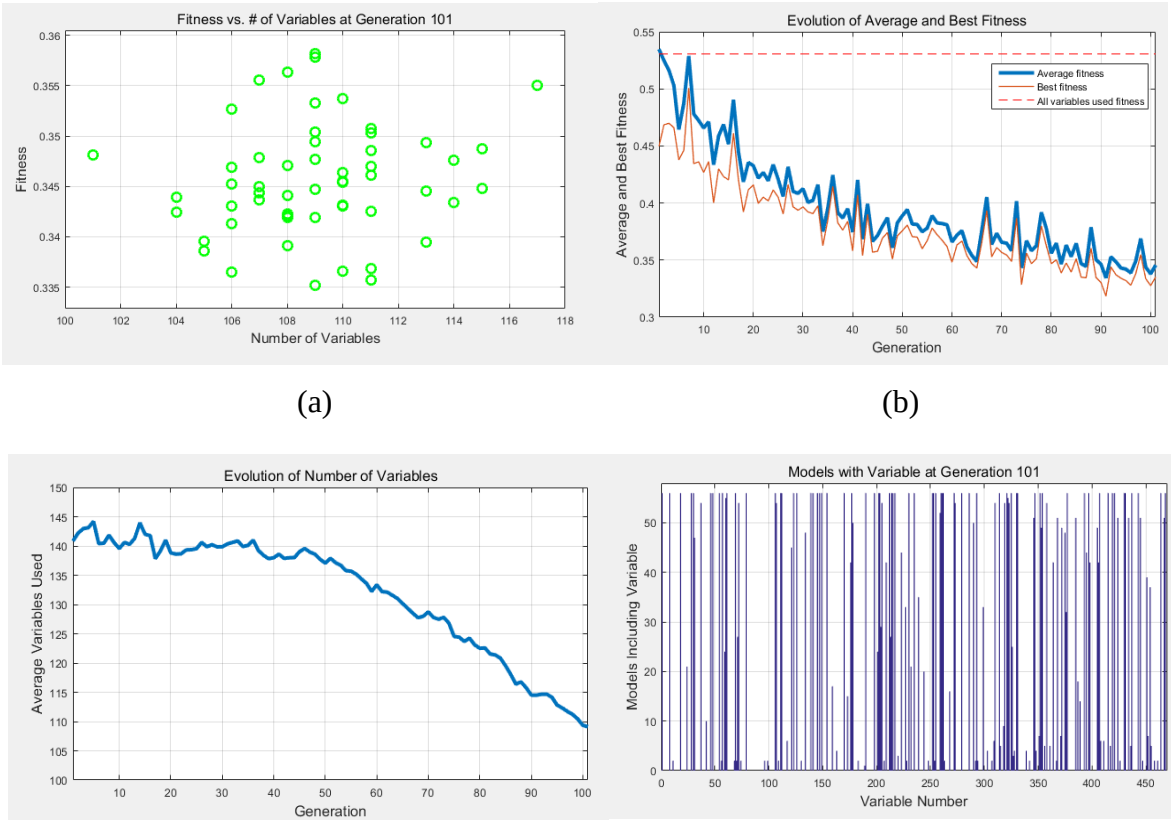


Figure 4-5 Variable selection result from PLSR-GA method
(Data sample collected from August 21-25 2016)

(a) Fitness and number of variable at final generation, (b) Trend of model fitness (*RMSECV*) in each generation, (c) Evolution trend of number of variables with generation, (d) Selected variable number at final generation

Figure 4-6 shows the feature selection results in twelve scenarios in summer season 2016. In the figure, 42 measurements from weather condition, chiller, AHU1, AHU2 and AHU3 are illustrated. The black block represents the measurement selected in each scenario. The feature selection results for the entire data measurement set for summer season and winter season can be found in Appendix VI.

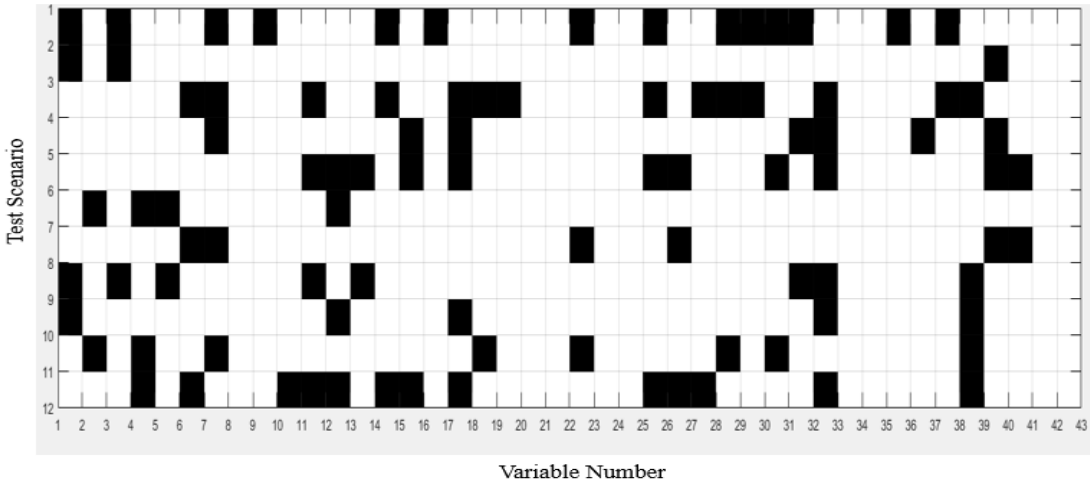


Figure 4-6 method key variable selection result (July to August, 2016)

4.4 Development of Weather/Schedule Information based Pattern Matching

4.4.1 Introduction of Data Pattern Matching

Similar to other process control system, a building HVAC system may operate differently under different external (e.g. weather) conditions and internal (e.g. occupancy) conditions. Hence a system's operation processes are time-varying due to the changes of a process's characteristics. Therefore, conventional monitoring strategy with unique system model does not show good prediction performance due to the self-limitations of these methods and the unique characteristics of multi operation processes such as time-varying, non-linearity, non-Gaussian and multiphase control [78].

A PM method is therefore needed to identify similar equipment operating conditions in historical database that are known to be fault free [38, 79, 80]. Compared with the monitoring and fault detection based on the traditional multivariate statistical process monitoring techniques which employ global system operation model, the PM strategy can identify and develop local operation model. Therefore, historical data can be better grouped to provide useful information and higher monitoring resolutions [81].

In this research, a weather/schedule based PM named as WPM method by using time series data mining technique is employed to locate similar system operation status and generate the WPM baseline data for the local PCA model.

4.4.2 SAX Method

In this research, SAX method is used to search similar weather time series data in historical database. SAX is a Piecewise Aggregate Approximation (PPA) based method to find similarity of time series data [82]. SAX, rather than other traditional PM techniques, is used in this study because of its ability to handle time-series data efficiently. A preliminary study (not included in this report) has demonstrated that traditional PM techniques have a significantly higher computation burden than the SAX technique when handling BAS data.

The PPA method is an equal sized segment time series data cutting technique. In the PPA method, the set of time series which constitute the database as $Y = \{Y_1, \cdots, Y_k\}$ is defined compared with the original time series dataset $X = \{x_1, \cdots, x_k\}$. Each sequence in X is n units long. Let N be the dimensionality of the transformed space and set the index ($1 \leq \mathrm{N} \leq \mathrm{n}$). Here, N is assumed as a factor of n [83]. Then, the original time series data X of length n can be represented in N space by a vector $\bar{X} = \bar{x}_1, \cdots, \bar{x}_k$. The ith element of $\bar{X}$ can be calculated by:

$$\bar{x}_i = \frac{\mathrm{N}}{n} \sum_{j=\frac{n}{N(i-1)}+1}^{\left(\frac{n}{N}\right)i} x_j \tag{10}$$

By this means, a time-series data is divided into N equal sized segment. At the same time the data with n dimensions can be reduced to N dimensions. Then, mean value of the data in each segment can be calculated. Therefore, this transformation produces a piecewise constant approximation of the original time series dataset. Figure 4-7 illustrates this transformation based on PPA method. The x axis represents the time-series and y axis represents the data range.

Similarly, in the SAX method, time series data is also divided into equal width data segments. These segments are represented by different symbolic strings. Through this process, the time series data dimension is reduced and similar data pattern can be easily marked as a discrete low dimension data for later clustering to find similar pattern. Finding similarity in a lower dimension data is more efficient. The test data and candidate historical fault free data can be combined to one L length time series $\boldsymbol{X}$(t). $\boldsymbol{X}$(t) is then normalized and divided according to normalized mean breakpoint. Different symbolic letters are assigned to the $\boldsymbol{X}$(t) according to the normalized data and clustering tool such as VizTree can be used to cluster the decomposed data [84].

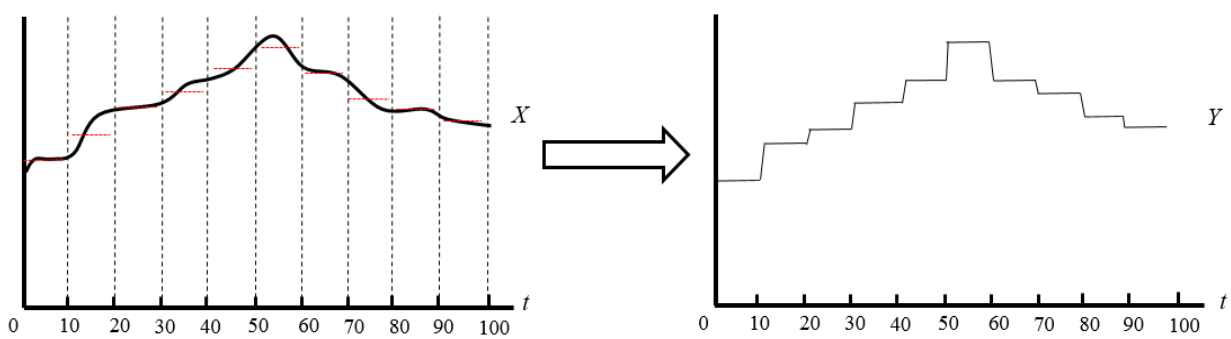


Figure 4-7 Illustration of PPA method

4.4.3 SAX Time Series Data Pattern Matching

For a HVAC system in buildings, the impacts from weather condition, occupants and different electrical appliances play a critical role in HVAC system operation. Usually, the schedule of

occupants and the usage of electrical appliances follows relatively fixed schedule and are seldom monitored in current practice. Therefore, finding and matching similar weather pattern to generate baseline data are main tasks in this study.

When using the SAX method, the historical baseline data should be firstly generated. Then the weather information data from historical baseline data and the new incoming snapshot data are combined together to produce one time series dataset. The incoming snapshot data's weather information are the head of this dataset. The dataset $\boldsymbol{X}$(t) is normalized by using z-score normalization as shown in:

$$x_{norm} = \frac{x - \bar{X}}{\sigma_x} \tag{11}$$

where, x is the time series data, $\bar{X}$ is the mean, σ_x is the standard deviation.

After normalization, dataset $\boldsymbol{X}$(t) is divided into $\boldsymbol{N}$ individual non-overlapping segments which have equal size data segment. One data segment is defined as one time window. In the research, same data window size is used for both snapshot dataset and historical dataset. The window size is determined by two factors: how fast weather conditions change and computation efficiency. If the window size is too large, weather has changed largely within a data window, which will reduce Pattern Matching accuracy. If the window size is too small, the computation burden will be increased. A number of alphabet size $\boldsymbol{a}$ is used to convert each $\boldsymbol{W}$ window into a SAX word. Alphabet size $\boldsymbol{a}$ is used to determine how many alphabet (letter) will be used to partition the subsequence of data. Each alphabet corresponds to a subsequence of data. The time series $\boldsymbol{X}$(t) is tagged by using different symbolic alphabet as shown in Figure 4-8.

In this research, outdoor enthalpy is used as the tagging target. The selection of enthalpy is because enthalpy information includes both temperature and humidity information. In Figure 4-8, one day outdoor enthalpy data is firstly normalized and divided into 48 equal-sized snapshot window (30-minute time elapse in each snapshot window). The selection of snapshot window size for WPM-FPCA method will be discussed in section 4.6.1. Then the data is tagged by using different symbolic characters as '*a*' to '*j*' and also marked with different color.

When the weather information time series dataset is tagged, the data section with same symbolic tag will be clustered to generate the WPM baseline dataset. In this research, a sample pool size $\boldsymbol{S}$ is used to define how many samples will be used to generate a baseline dataset. Although a smaller window size can accurately capture the weather change, less information is obtained to generate a valid baseline database if a small window size is used to partition the data. Therefore, an adjacent dataset to the test window can be used as the sample pool because the building occupancy may remain steady in a relatively long period.

Figure 4-9 summarizes the above-discussed procedure of WPM by using SAX method. Figure 4-10 demonstrates a result of the WPM process by using outdoor air enthalpy information. It can be seen that when implementing the SAX based WPM, outdoor air enthalpy information obtained in each snapshot window can find its match from historical baseline data. Therefore, system operation status under similar weather conditions can be identified and the WPM baseline dataset for each snapshot window can be efficiently generated for developing PCA models used in the following fault detection process.

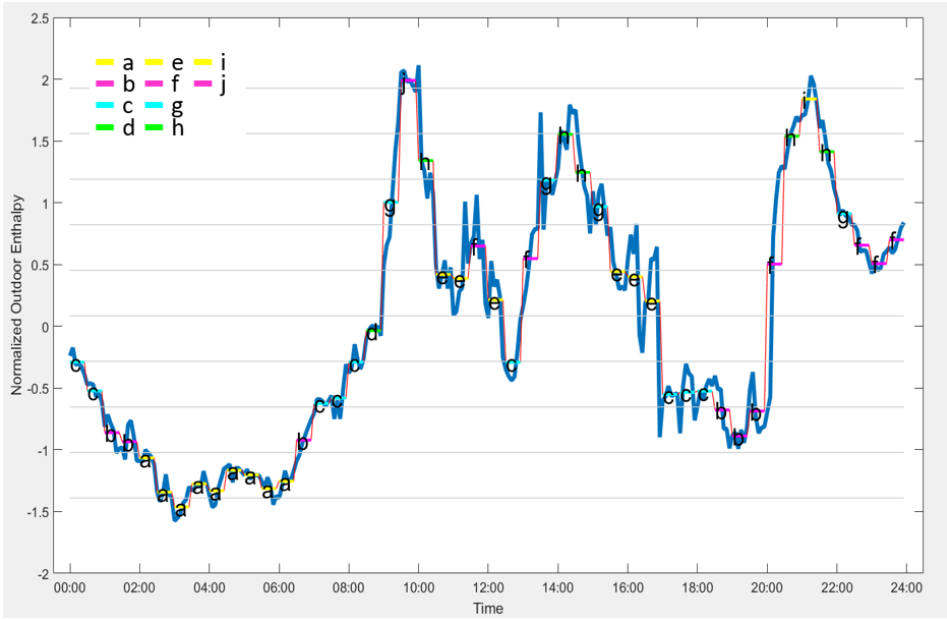


Figure 4-8 SAX word in tagging enthalpy condition

4.5 Development of PCA for Building Fault Detection

4.5.1 Introduction

PCA is one of the most popular multivariate statistical process control (MSPC) tools employed in process control industry to monitor system operation [85]. PCA is used for multivariate data when the correlation among the large number of variable measurement sets is difficult to be described using physical modeling method. By projecting the data into a lower-dimensional space, the high dimensionality of system operation process variables can be more accurately represented. Through implementing PCA, a lower-dimensional representation which preserves the correlation structure among a large number of system variables can be obtained and the variability in the data can be clearly captured [86].

In the past two decades, the PCA method has been widely used for FDD in many fields including building equipment fault detection. Wang et al., employed PCA in AHU sensor fault detection [87], and later expand this method to detect VAV terminals [88] and central chilling systems [89]. Although PCA tool has been demonstrated to be useful in some equipment fault detection, some weaknesses are found to hinder its application when coping with more complicated conditions such as if some error samples are included when training PCA model, system propagation characteristic, and system sample noise. In order to solve these issues, several updated PCA tools are developed. For example, Hu et al., developed a self-adaptive PCA for chiller sensor fault detection [90]. Du et al., improved the PCA tools with joint angle analysis to detect faults in the VAV system [91]. Xu et al., combined wavelet analysis and PCA method to improve the chiller sensor fault detection result [92]. Li et al., proposed a PM-PCA for AHU system fault detection [38].

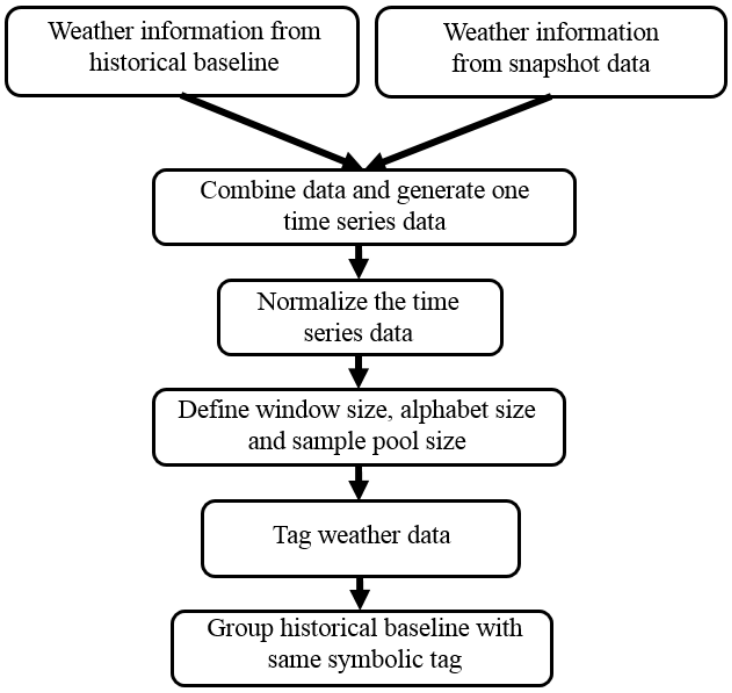


Figure 4-9 Flow of WPM by using SAX

Figure 4-10 Illustration of WPM by using SAX

However, there is no report in the existing literature using MSPC tools such as PCA in whole building system fault detection. In this research, PCA is employed in the multi-structure fault detection strategy for whole building fault detection after implementing system feature selection and system operation PM.

4.5.2 PCA Method

PCA uses orthogonal transformation procedure to extract a set of linearly uncorrelated principal components (PCs) from the possibly correlated original variables [93]. In PCA [94], a data measurements matrix $\boldsymbol{X} \in R^{n\times m}$ includes n samples and m process variables. Measurement $\boldsymbol{X}$ can be decomposed into a principal component (PC) matrix $\widehat{\boldsymbol{X}}$ (also known as PC subspace) plus a residual matrix $\boldsymbol{E}$ (also known as residual subspace). In the principal matrix $\widehat{\boldsymbol{X}}$, system process variations are captured. In the residual matrix $\boldsymbol{E}$, system noise and error information can be mainly captured. The principal matrix $\widehat{\boldsymbol{X}}$ is a product of a scores matrix $\boldsymbol{T}$ ($I \times A$) and a loadings matrix $\boldsymbol{P}$, as given in equation:

$$\boldsymbol{X} = \hat{X} + E = TP^T + E = \sum_{i=1}^{a} t_i p_i^T + E \tag{12}$$

where t_i is a score vector (orthogonal) which contains information about relationship between samples and p_i is a loading vector (orthonormal) which contains information about relationship between variables. a is the number of PCs which explain the variability of a process. Figure 4-11 shows the decomposition of measurement matrix $\boldsymbol{X}$.

When implementing PCA, the following procedures are usually carried on [95]:

Firstly, the sample data matrix X is scaled. Columns of matrix X are mean centered by subtracting the mean of each column from each entry in the column. Then, a covariance matrix of X can be obtained [96]:

$$cov(X) = \frac{X^T X}{m-1} \tag{13}$$

where m is the number of samples.

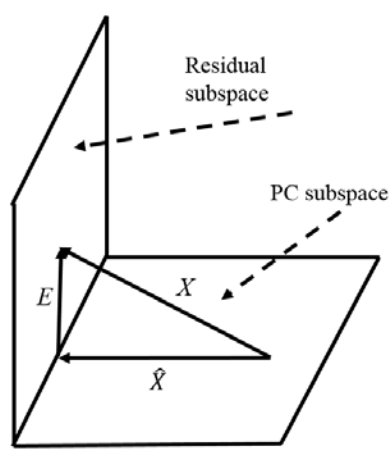


Figure 4-11 Decomposition of measurement matrix X

The next step is to calculate the principal component. Usually, two algorithms such as singular value decomposition (SVD) and Non-linear iterative partial least squares are employed to calculate

the principal component. Here, we use SVD algorithm to decompose data measurement matrix X (m by n, m sample and n variable) as:

$$X = U\sum V^T \tag{14}$$

where U (m by m) and V (n by n) are orthonormal matrixes, $\sum$ (m by n) is a diagonal matrix. The elements of S are defined as singular value σ and place in $\sum$ in a descending order. Here, each singular value σ is the square root of the corresponding eigenvalue of the covariance matrix as $\sigma_i = \sqrt{\lambda_i}$. The first PC is associated with the largest singular value. In PCA, the loadings matrix P are identical to the matrix V as:

$$P = V \tag{15}$$

The scores matrix T can also be represented by SVD parameters U and $\sum$ as:

$$U\sum = T$$

Therefore, Equation 12 can also be written as:

$$X = \sum_{i=1}^{a} t_i p_i^T + E = U_a \sum_a V_a^T + E \tag{16}$$

The next step is to choose the number of PCs retained in the model. Although, there is no best strategy to determine how many PCs are retained in the model, there are some other rules of thumb and user's knowledge used to determine the number of PCs [77]. Two methods of selecting number of PCs retained in the model are introduced and used in later PCA fault detection.

One method is to look at the plot of eigenvalues (or known as scree plot method) [97]. Scree plot exhibits the eigenvalues in each PC in a descending order as shown in Figure 4-12. When there is an obvious "turning point" of eigenvalue founded in the scree plot, the numbers of PC corresponding to that "turning point" are retained and the rest of PCs is discarded. For example, in Figure 4-12, "turning point" is found in PC No. 5 which the eigenvalue is 0.9. Therefore, the first five PCs are retained in the model.

Another method is cumulative variance percentage contribution method. In this method, the smallest number of PCs is selected to capture a certain percentage of the cumulative variance which is pre-determined by the users. This number of PCs contributes the maximum cumulative variance [97].

$$\frac{d_1 + d_2 + \cdots + d_q}{d_1 + d_2 + \cdots + d_p} \geq \theta \tag{17}$$

where θ is the pre-determined percentage, d is variance captured by each PC, p is the total number of PC, q is the retained number of PC.

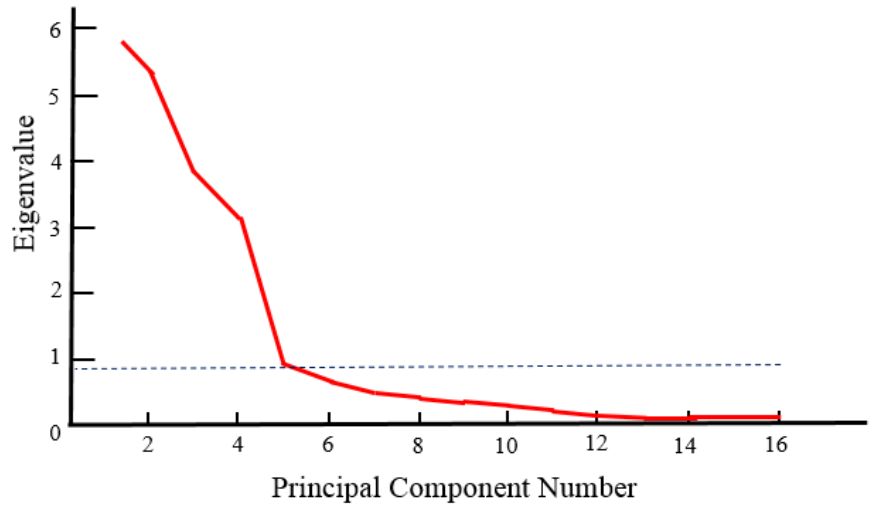


Figure 4-12 Number of PCs determination by using scree plot

4.5.3 Statistics for PCA Fault Detection

In process monitoring and fault detection, Hotelling statistics (T^2) and Squared Prediction Error (*SPE*) [95] statistics are commonly used with PCA techniques to judge whether the operation is under abnormality or not. When a fault occurs in an operation process, such fault usually causes one or several changes in monitored variables, resulting in some PCs with larger variation which are beneficial for monitoring the fault (informative PCs), and other PCs with less or without variation.

T^2 can be calculated for each new observation by:

$$\boldsymbol{T^2} = x^T P \sum a \, P^T x \tag{18}$$

where $\sum a$ contains the non-negative eigenvaluescorresponding to the a principal components.

The upper confidence threshold of T^2 can be calculated by using F-distribution as:

$$\boldsymbol{T^2_{threshold}} = \frac{a(n-1)}{n-a} F_{a,n-a,\alpha} \tag{19}$$

where n is the number of samples in the data, α is the level of significance.

Another statistic is SPE value which can be calculated as following:

$$SPE = ||(I - PP^T)x||^2 \leq {Q_\alpha}^2 \tag{20}$$

where Q_α is SPE control limit which can be calculated as follows:

$$Q_\alpha = \theta_1 [\frac{c_\alpha \sqrt{2\theta_2 h_0^2}}{\theta_1} + 1 + \frac{\theta_2 h_0 (h_0 - 1)}{\theta_1^2}]^{\frac{1}{h_0}} \tag{21}$$

where

$$\theta_i = \sum_{j=k+1}^{n} \lambda_j^i$$

$$h_0 = 1 - \frac{2\theta_1\theta_3}{3\theta_2^2}$$

The *SPE* statistic is a measure of the amount of variation not captured by the PCA model. T^2 statistic is a measure of the variation in each sample within the PCA model. It indicates the distance of each sample from the center of the model [95]. As the *SPE* statistic is more sensitive to detect faults with small magnitudes [98], when implementing a whole building fault, SPE tends to generate a high false alarm rate. Therefore, in this study, T^2 statistic is used to obtain the detection result.

4.5.4 Flow of PCA Fault Detection

When implementing the PCA method for fault detection, the following procedures are followed: firstly, WPM baseline dataset, i.e., data that have similar weather condition and schedule the snapshot data, will be identified from the baseline dataset using the SAX based WPM method. Two PCA models will be developed, one for the WPM baseline dataset and one for the snapshot data. Statistic thresholds will be established for the WPM baseline dataset. Statistical value, i.e. T^2 from the snapshot data, will be compared with the threshold. Fault is flagged if the T^2 value (from snapshot data) is larger than the T^2 threshold (from the WPM baseline dataset). Figure 4-13 shows the flowchart of a WPM-PCA fault detection process. In Chapter 6, this WPM-PCA method will be evaluated using real building data.

4.6 Parameter Sensitivity Test for WPM-FPCA Fault Detection

When implementing the SAX method for WPM, there are several parameters including snapshot window size and sample pool size that need to be determined. In order to evaluate the effects of different parameter values on the PM and fault detection performances, sensitivity tests need to be performed. In this study, the sensitivity tests on snapshot window size, data sample searching pool size are performed and are summarized below.

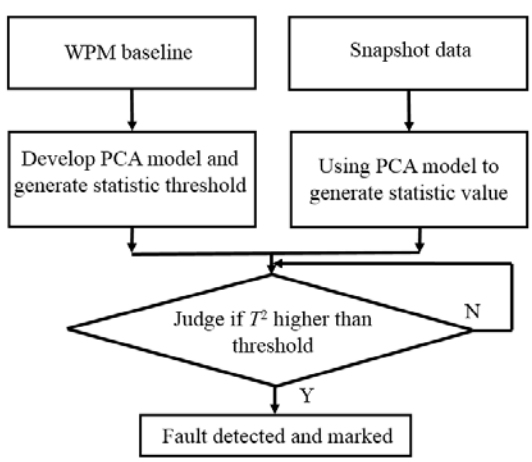


Figure 4-13 Flowchart of WPM-FPCA for fault detection

4.6.1 Sensitivity Test on Snapshot Window Size

In order to identify the effects of window size on the PM and fault detection result, sensitivity tests have been performed. Three window sizes, i.e., 15-minute, 30-minute and 45-minute time window size have been used to test the PM performance. The PM performance is evaluated by 1) examining the weather pattern matching result, and 2) examining the fault detection result by using fault test day and free day. Data from the summer of 2016 have been used to implement the sensitivity test. Historical data that provide a 30 day baseline dataset is used in each test. Five days are used as the test days. A WPM dataset with smaller window size helps SAX method to capture the weather dynamic change better, and therefore generate a more accurate baseline data. This would further help to better detect faults with higher dynamic responses under different weather and indoor conditions.

Figure 4-14 shows the PM results under the three window sizes for test day of July 6th, 2016. From the Figure, it can be observed that smaller window sizes (15-minute and 30-minute) produce more accurate baseline data than the 45- minute window size.

Five randomly selected fault test days (August 8th, September 7th, September 11th in 2016, and July 9th, July 11th in 2017) are used to evaluate the performance of the fault detection method under different time window sizes. The comparison also shows that the baseline data generated under smaller window sizes can produce a more sensitive detection result as indicated by the number of abnormal samples (automatically counted in the post-process). For example, in September 7th 2016, T^2 value in 72% of the total time during the fault test overpasses the threshold under 15-minute snapshot window size, compared with 41% and 33% using 30-minute and 45-minute snapshot window size respectively, as shown in Figure 4-15.

However, a smaller window size may also cause an increased false alarm rate in some cases. For example, Figure 4-16, which is generated by using a November 1st 2016, shows the fault detection comparison for a fault free day. It can be observed that in this case, the fault detection method in the 15-minute window size yields false alarm for some periods. No false alarm is observed in the 30-minute window size and 45-minute window size cases.

Based on the above analyses, in the following fault detection test, a window size of 30-minute is adopted to reach a trade-off between detection accuracy and false alarm rate.

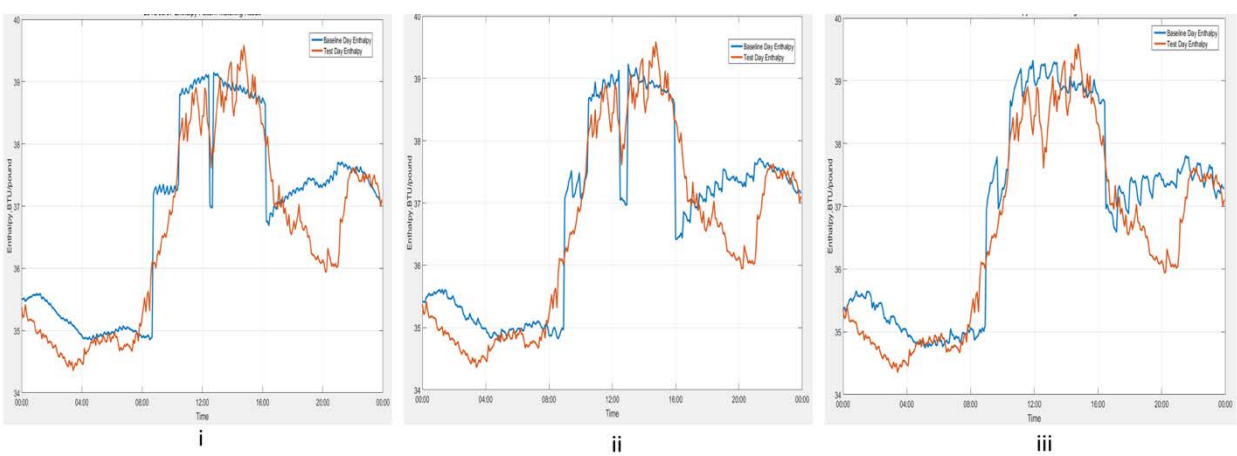


Figure 4-14 PM comparison under different window sizes (i. window size 15-minute, ii window size 30-minute, iii. window size 45-minute)

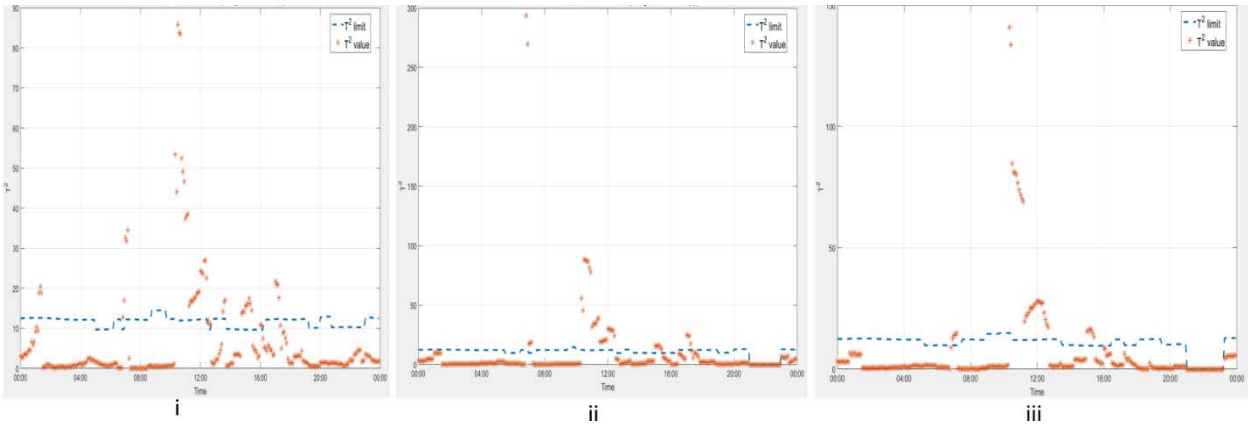


Figure 4-15 Fault detection comparison under different window sizes (i. window size 15-minute, ii window size 30-minute, iii. window size 45-minute)

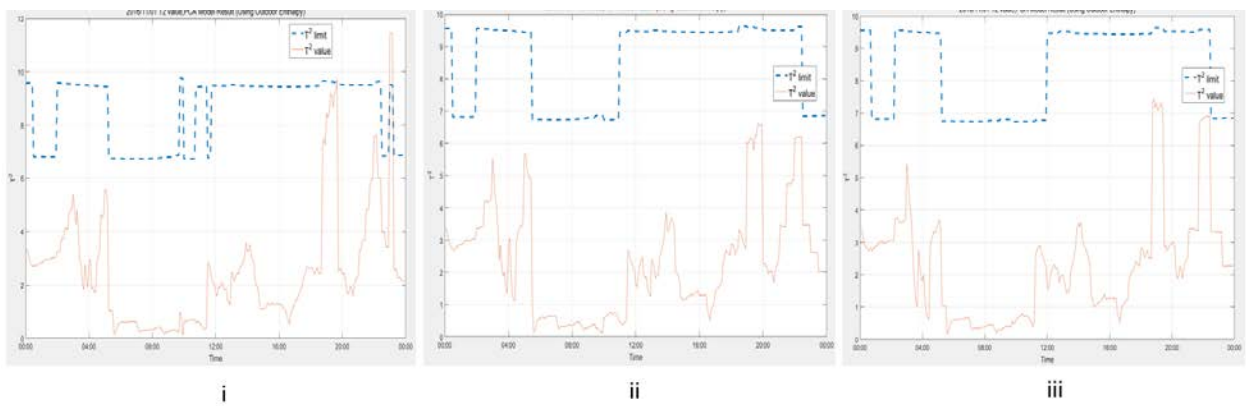


Figure 4-16 Fault detection comparison under different window sizes (i. window size 15-minute, ii window size 30-minute, iii. window size 45-minute)

4.6.2 Sensitivity Test on Baseline Data Size and Sample Searching Pool Size

The literature does not have suggestions as to how many data samples should be included when generating a PCA model for fault detection. In this study, a minimum of 100 data samples is set as the goal for the WPM baseline dataset when developing a PCA model. In order to meet this requirement, two parameter sensitivity tests have been implemented.

Firstly, we investigated how many historical baseline days should be collected for the baseline dataset. Obviously, including more fault free days in the baseline dataset will increase the accuracy and robustness of the fault detection method. However, in the real practice, especially for building re-commissioning, it may not be possible to obtain a large number of fault free days. In this investigation, a 20-day fault free baseline dataset and a 30-day fault free baseline dataset are formed for each season and used in the sensitivity analysis.

To evaluate these two historical datasets, 30 days testing data from summer (July), 30 days from fall (November) and 30 days from winter (December) of 2016 have been employed. These testing data do not have any overlap with the historical baseline data. The searching pool size is

fixed as 120- minute (2 windows before the snap shot window and 2 windows after the snap shot window) for this sensitivity test. The impact of the searching pool size will be evaluated next.

Table 4-2 summarizes the maximum sample number, minimum sample number and average sample number in the WPM baseline dataset for each tested snapshot window. At the same time, the percentages of the snapshot windows which finds less than 100 data samples in its WPM baseline dataset over all the snapshot windows are calculated. This percentage is referred to as the "small sample size percentage" hereafter. The higher this percentage is, the more the tested snapshot windows do not find enough similar fault free baseline data.

Table 4-2 Sensitivity analysis on fault free date searching number

Test Period	**Maximum Number of Sample**		**Minimum Number of Sample**		**Average Number of Sample**		**Percentage of Windows with Sample Size Less than 100**	
Searching Date Number	20	30	20	30	20	30	20	30
Summer	216	288	0	0	77	120	72%	38%
Fall	216	282	0	0	72	106	75%	49%
Winter	216	312	0	0	88	123	60%	36%

The results show that, all of the average sample numbers in each WPM baseline dataset are less than 100 for all tested seasons when using the 20-day baseline dataset. When using the 30-day baseline dataset, this number, i.e., the average sample numbers in each WPM baseline dataset, increases to be more than 100 for all tested snapshot windows. The small sample size percentage reaches 72%, 75% and 60% respectively for each test season when using the 20-day fault free baseline dataset. These small sample size percentages was decreased to be 38%, 49% and 36% when using the 30-day fault free baseline dataset. Therefore, the 30-day fault free baseline dataset is used in the future fault detection studies.

After the size of the historical baseline dataset is selected as 30 days, the size of the data searching pool size is evaluated next.

When forming the WPM baseline dataset, neighboring windows of a snapshot window is typically included in the searching pool for pattern matching. For example, when analyzing a snapshot window of 10:00am-10:30am, all collected historical baseline data between 10:00am-10:30am, as well as its neighboring windows, such as 9:30am-10:00am and 10:30am-11:00am, can be used to form the searching pool, among which, those with similar weather will be selected as the WPM baseline dataset, as shown in Figure 4-17. This searching pool size (S) will affect the WPM accuracy. If S is too small, there would not be enough samples to be included in the WPM baseline dataset. If S is too large, the time frame of the neighboring windows will be too long. In the above example, when extending the concept of neighboring window to be from 8:00am - 12:00pm, instead of 9:00 -11:00am, data that contain very different internal loads would be included in the searching pool, and thus affect the WPM accuracy In this sensitivity study, three data sample searching pool sizes, namely, 120- minute (2 windows before and 2 windows after), 180-minute (3 windows before and 3 windows after) and 240-minute (4 windows before and 4 windows after) are considered. Here, the snapshot window size is chosen as 30-minute, based on

a previous sensitivity test. The testing data and evaluation criteria used here are similar to those in the historical baseline database size evaluation.

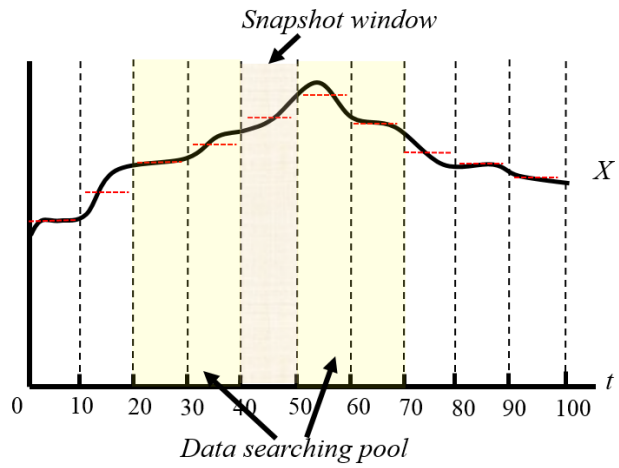


Figure 4-17 Illustration of data searching pool

Observing the results, it can be concluded that the data sample number increases with the increase of the searching pool size. When using the 120-minute searching pool size, the average number of data sample reaches 120, 106 and 123 for the summer, fall and winter season, respectively. When using the 180-minute searching pool size, this number increases to be 179, 159 and 184.

It can also be observed that the number of data sample is relatively low in the fall season (data from November, 2016) when compared with other two seasons (i.e., data from July and December, 2016). It is partly because the weather changes quickly over a day in the fall season in Philadelphia. So, it is more difficult for the WPM method to find similar weather pattern

Table 4-3 summarizes the test results for each data sample searching pool size in different seasons.

Observing the results, it can be concluded that the data sample number increases with the increase of the searching pool size. When using the 120-minute searching pool size, the average number of data sample reaches 120, 106 and 123 for the summer, fall and winter season, respectively. When using the 180-minute searching pool size, this number increases to be 179, 159 and 184.

It can also be observed that the number of data sample is relatively low in the fall season (data from November, 2016) when compared with other two seasons (i.e., data from July and December, 2016). It is partly because the weather changes quickly over a day in the fall season in Philadelphia. So, it is more difficult for the WPM method to find similar weather pattern.

Figure 4-18 to Figure 4-20 show the frequency of the size (data sample number) of the WPM dataset for snapshot windows in different seasons. From the frequency distribution, it can be seen that in fall and winter seasons, the frequency of snapshot windows with smaller “similar baseline data” sample size increases dramatically. It means that it is more difficult for the WPM method to find similar weather pattern. Again, this could be due to the dynamic daily weather pattern. Therefore, different data searching pool size can be considered for different seasons in the future research.

From the analysis above, the 180-minute data sample searching pool size has met the requirement of at least 100 data samples in each snapshot window. Hence, the 180-minute data sample searching pool size is used in WPM-FPCA fault detection method.

Table 4-3 Sensitivity test on data searching pool size

Test Period	**Summer**			**Fall**			**Winter**		
Data searching pool size	120-minute	180-minute	240-minute	120-minute	180-minute	240-minute	120-minute	180-minute	240-minute
Maximum Number of Sample	288	408	501	282	432	570	312	456	588
Minimum Number of Sample	0	0	0	0	0	0	0	0	0
Average Number of Sample	120	179	238	106	159	213	123	184	246
Percentage of Window with Sample Number Less than 100	38%	20%	12%	49%	27%	20%	36%	23%	15%

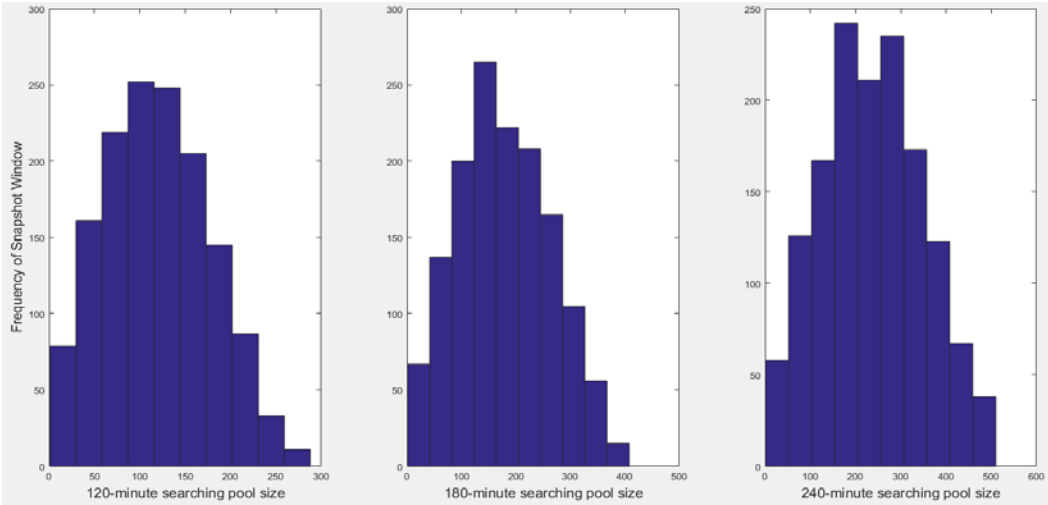


Figure 4-18 Frequency of number of data sample for the 30 test days in summer season

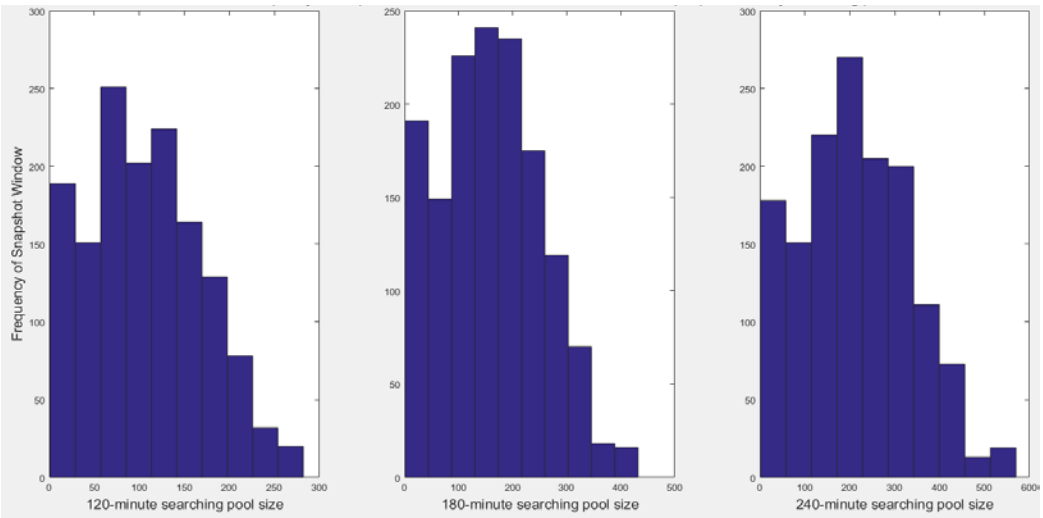


Figure 4-19 Frequency of number of data sample for the 30 test days in fall season

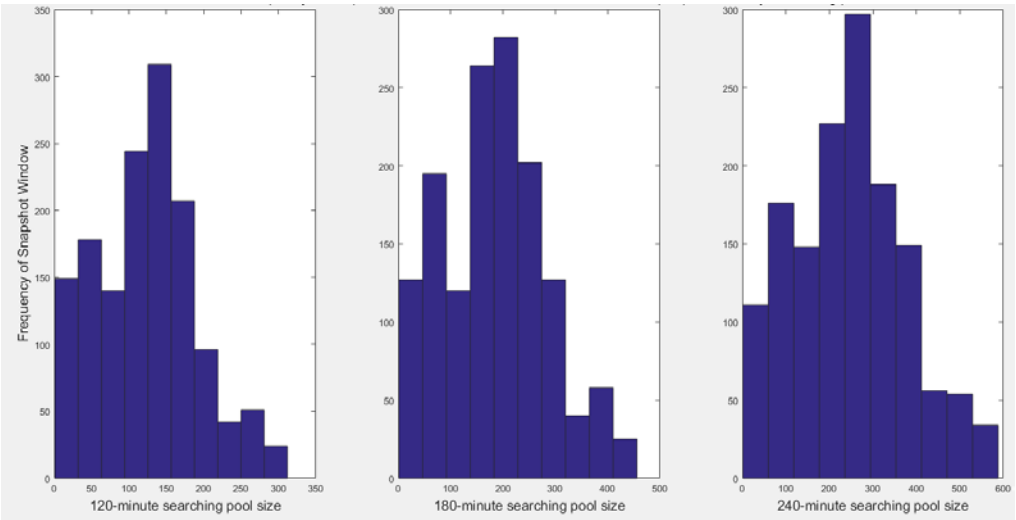


Figure 4-20 Frequency of number of data sample for the 30 test days in winter season

4.7 Summary

In this Chapter, a data-driven method named as WPM-FPCA method is developed for whole building fault detection. Firstly, historical baseline data are collected that reflect an acceptable building status. A PLSR-GA algorithm is then used to identify key system variables (features) from all available variables in the historical baseline dataset. These key features which represent system operation are used for developing the following PCA models. Secondly, a WPM method is developed based upon the time series SAX algorithm. The WPM method can efficiently find a WPM baseline dataset that has similar weather and building conditions with the incoming snapshot data with minimum computational burden, which makes it suitable for future on-line fault detection for large and complex building system. Lastly, PCA models are developed for both the WPM baseline dataset and snapshot data and are used for an automated whole building fault

detection process. Parameter sensitivity tests including snapshot window size and data sample searching pool size are implemented to find the impacts of the parameters from the developed methods. Proper parameters are then selected and used. The evaluation process of the developed method is described in Chapter 6.

Chapter 5 Development of Fault Diagnosis Method

5.1 Introduction

When a whole building level abnormality is detected, the root-causes of this abnormality need to be diagnosed and located so that these faults can be corrected.

In this research, a Bayesian Network (BN) based method is proposed to diagnose and isolate root-causes for whole building faults. BN based methods have been successfully developed for FDD and system maintenance in different areas because BN has demonstrated an excellent ability to work well under uncertainty and incomplete information [99].

In a building HVAC system, there are existing studies that have adopted the BN for component level fault diagnosis such as chiller, AHU, and VAV terminals [100-104]. For example, Zhao et al., developed diagnostic BN for AHU and intelligent chiller FDD method by using a three-layer Bayesian belief network [100]. They also developed diagnostic BNs to diagnose component faults in AHU [101, 102]. Xiao et al., proposed a BN based FDD strategy for diagnosing ten typical faults of VAV terminal unit [103]. Although these studies demonstrate good potentials of using BN in component level fault diagnosis, there is a lack of study that uses BN diagnosis methods for whole building faults where interactions exist in different subsystems.

When applying BN based fault diagnosis method for building system faults, the following challenges exist: 1) it is impractical to develop a system-level BN from exhausted fault data by implementing fault experiments in a real building. Hence, determining parameters (prior and conditional probabilities) are challenging; 2) due to the coupling impacts, fault root-cause determination and fault evidence are often more complicated for whole building faults than component-level faults; and 3) it is harder to develop baseline models for whole building faults due to the large dimension of whole building dataset.

In this research, expert and physical knowledge are used to develop a BN structure model and part of the BN parameters. The WPM method [105] which has been employed in the WPM-FPCA fault detection method is used to create a baseline data for evidence generation and to obtain *LEAK* probabilities which are parameters needed in the proposed BN method.

5.2 Bayes Theorem

A BN is based on Bayes theorem. In Bayes theorem, conditional probability distribution is used to measure the probability of an event under the assumption that another event has occurred. For example, if an event **A** occurs (here **A** may represent a fault) when event **B** (here **B** may represent a symptom) is known or assumed to have occurred, the probability of **A** under the condition **B** can be written as [106]:

$$P(A|B) = \frac{P(AB)}{P(B)} = \frac{P(B|A)P(A)}{P(B)} \tag{22}$$

where P(AB) is the joint probability of events A and B, and $P(AB) = P(B)P(A|B) = P(A)P(B|A)$.

Supposing A1,A2, . . . , An are a set of random variables and satisfy: (a) $\sum_{i=1}^{n} A_i = S$, where S is the certain event; (b) they are mutually independent; and (c) P(Ai) > 0, i= 1,2,. . . ,n, for any given event B, the following marginal probability can be obtained:

$$P(\mathrm{B}) = \sum_{i=1}^{n} P(A_i)P(B|A_i) \tag{23}$$

Through this way, Equation 22 can be re-written as:

$$P(A_i|B) = \frac{P(AB)}{\sum_{i=1}^{n} P(A_i)P(B|A_i)} = \frac{P(B|A_i)P(A_i)}{\sum_{i=1}^{n} P(A_i)P(B|A_i)} \tag{24}$$

A Bayesian inference is used to find out the cause of an event from the effects of an event by calculating the posterior probability. Bayesian theorem provides a way to calculate the posterior probability (left side in Equation 24) from the prior probabilities (right side in the equation). The prior probability of variable $\boldsymbol{A_i}(\boldsymbol{P}(\boldsymbol{A_i}))$ and the conditional probability of the variable B given $\boldsymbol{A_i}(\boldsymbol{P}(\boldsymbol{B}|\boldsymbol{A_i}))$ firstly assigned through the existing knowledge of the problem which can be either some statistic results or the estimation from the experts in the field. Then, the posterior probability $\boldsymbol{P}(\boldsymbol{A_i}|\boldsymbol{B})$ can be calculated via Equation 24.

The BN includes two elements: structure model and parameters. Nodes and arcs are used in a BN structure model to describe relationships among events. Each node represents a variable which is assigned to a probability. Arcs are the direct reasoning connection between nodes. Once the BN structure model is defined, the BN parameters (including prior probability distribution, conditional probability distribution and *LEAK* probability distribution) for each node should be decided to generate a Bayesian inference.

5.3 Development of BN Structure Model

5.3.1 BN Structure

The development of a BN structure model is to identify the cause-and-effect relationship. Usually, expert based knowledge is used to develop the system rule which will be mapped to the BN. During the development process, the first step is to determine the network layer according to the nature of the problem. Traditionally, a two-layer BN structure which includes a fault layer and a fault evidence layer are employed to develop BN for fault diagnosis problems. Apart from the two-layer BN structure, more layers may be added to reflect the system behavior in more details due to the complexity of the system. For example, an additional information layer was used to develop BN for chiller and AHU fault diagnosis [100-102]. In this research, a two-layer BN structure is designed.

5.3.2 BN Node

The second step in developing a BN structure model is to assign BN nodes. BN nodes are assigned to represent different variables through system rule mapping or knowledge representation.

In this study, two types of BN nodes are need to be determined as fault node and fault evidence node.

Fault nodes represent whole building level faults root causes. For example, an AHU supply air temperature sensor bias fault (either positive bias or negative bias) is assigned to one fault node. The states of the fault node are divided into two types as faulty and fault free.

Evidence nodes represent observable fault symptoms. Fault symptoms typically come from two sources:

1) Concurrent relationships among measurements. For example, when an outdoor damper is stuck at 0% open position, the mixed air temperature measurement is the same as the return air temperature measurement, and is very different from the outdoor air temperature measurement;

2) Historical relationship of a measurement between its current value and its historical baseline value. For example, another symptom of the outdoor damper stuck at 0% open position fault is that the cooling coil valve opening will be smaller than its historical baseline values under similar weather conditions.

BN network is expressed graphically (Figure 5-1), in which, arcs are connected between fault nodes and evidence node to represent the cause-and-effect relations between them. For example, Figure 5-1 shows a BN for diagnosing AHU outdoor air damper stuck at a higher than normal position fault. This fault is assigned to a fault node named as AHU-OA-DMPR-Stuck-H. The fault evidences (also known as fault symptoms) in summer season are assigned as evidence nodes including chilled water flowrate (CHW-Flowrate), AHU cooling valve position (AHU-CC-VLV), the difference between AHU mixed air temperature and outdoor air temperature (AHU-MAT-OAT) readings, and chiller cooling energy (CHW-Cooling). These evidence nodes are connected to the fault node by adding arcs.

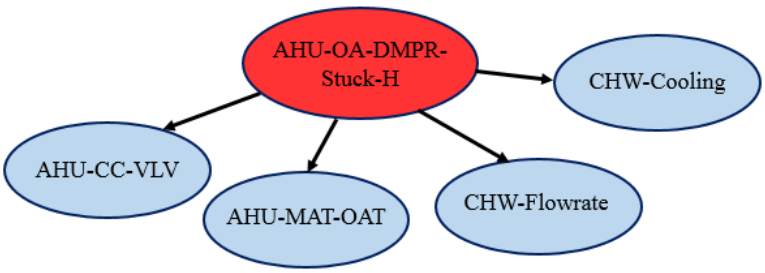


Figure 5-1 BN structure model example

5.3.3 Development of BN Structure for Whole Building Fault

Nine fault root causes are identified as whole building faults as summarized in

Table 5-1. Fault evidences, i.e., key measurements and virtual measurements (combination of measurements) used to observe a fault's symptoms, are summarized in Table 5-2.

Figure 5-2 demonstrates the developed BN structure model for all of the whole building faults considered in Table 5-3. Relationships between a fault and its symptoms are summarized in Table 5-3.

Table 5-1 Whole building fault list

Fault Category	Fault Name	Abbreviation
Primary cooling subsystem fault	Chilled water supply temperature sensor negative bias (screen reading lower than real value)	CHWS-Temp-Bias-N
	Chilled water supply differential pressure sensor positive bias (screen reading higher than real value)	CHWS-DP-Bias-P
Supply air subsystem fault	AHU outdoor air damper stuck at a higher than normal position	AHU-OA-DMPR-Stuck-H
	AHU supply air temperature sensor negative bias (screen reading lower than real value)	AHU-SA-Temp-Bias-N
	AHU cooling coil valve stuck at a higher than normal position	AHU-CC-VLV-Stuck-H
Operator Fault	Schedule fault (system is occupied while under normal operation, it should be unoccupied)	OpF-Sch-Occ
	Schedule fault (system is occupied while under normal operation, it should be occupied)	OpF-Sch-Unocc
	AHU cooling coil valve control override at a higher than normal position	OpF-AHU-CC-VLV-SWO-H
	Chiller is off while under normal operation, it should be on	OpF-Chiller-Off

Table 5-2 Fault evidence list

Subsystem	Evidence No.	Key Measurement and Virtual Measurement	Abbreviation
Primary cooling subsystem	E1	Chilled water supply temperature	CHWS-Temp
	E2	Chilled water return temperature	CHWR-Temp
	E3	Chilled water flowrate	CHW-Flowrate
	E4	Calculated chiller cooling load	CHW-Cooling
	E5	Chiller pump speed	CHW-Pump-Speed
Supply air subsystem	E6.1	AHU-1 outdoor air damper position	AHU1-AHU-OA-DMPR
	E6.2	AHU-2 outdoor air damper position	AHU1-AHU-OA-DMPR
	E7.1	AHU-1 cooling coil valve position	AHU1-AHU-CC-VLV
	E7.2	AHU-2 cooling coil valve position	AHU2-AHU-CC-VLV
	E8.1	Difference between AHU-1 mixed air and outdoor air temperatures	AHU1-AHU-MAT-OAT
	E8.2	Difference between AHU-2 mixed air and outdoor air temperatures	AHU2-AHU-MAT-OAT
	E9.1	AHU-1 supply air fan speed	AHU1-AHU-SF-Speed
	E9.2	AHU-2 supply air fan speed	AHU2-AHU-SF-Speed
	E10.1	AHU-1 supply air temperature	AHU1-AHU-SA-Temp
	E10.2	AHU-2 supply air temperature	AHU2-AHU-SA-Temp
	E11	VAV2 reheat coil valve (sum of reheat coil valve position in the VAVs which connect to AHU2)	VAV2-RHC-VLV (in winter season)
	E12	AHU-2 preheat air temperature	AHU2-PreHC-VLV(in winter season)

Table 5-3 Fault and evidence list

Fault	Fault Node	Symptom Description
1	OpF-Sch-Occ	E1(lower than normal); E2 (lower than normal); E3 (higher than normal); E4(higher than normal); E9.1 (higher than normal); E9.2 (higher than normal); E10.1(lower than normal); E10.2(lower than normal)
2	OpF-Sch-Uocc	E1(higher than normal); E2(higher than normal); E6.1(lower than normal); E6.2(lower than normal); E9.1(lower than normal); E9.2(lower than normal); E10.1(higher than normal); E10.2 (higher than normal)
3	OpF-AHU-CC-VLV-SWO-H	E3(higher than normal); E4(higher than normal); E7.1(higher than normal); E7.2(higher than normal); E10.1(lower than normal); E10.2(lower than normal)
4	OpF-Chiller-Off	E1(higher than normal); E2(higher than normal); E3(lower than normal); E4(lower than normal); E7.1(higher than normal); E7.2(higher than normal); E8.1(lower than normal); E8.2(lower than normal); E10.1(higher than normal); E10.2(higher than normal)
5	CHWS-Temp-Bias-N	E2 (higher than normal); E3(higher than normal); E4(higher than normal); E5(higher than normal); E7.1(higher than normal); E7.2(higher than normal)
6	AHU2-AHU-OA-DMPR-Stuck-H	E3(higher than normal) ; E4(higher than normal); E7.2(higher than normal); E8.2 (lower than normal); E11(higher than normal, in winter season) ; E12(higher than normal, in winter season)
7	AHU1-AHU-SA-Temp-Bias-N	E3(lower than normal); E4(lower than normal); E7.1(lower than normal); E9.1(higher than normal)
8	AHU2-AHU-SA-Temp-Bias-N	E3(lower than normal); E4(lower than normal); E7.2(lower than normal); E9.2(higher than normal)
9	AHU2-AHU-CC-VLV-Stuck-H	E3(higher than normal); E4(higher than normal); E7.2(lower than normal); E10.2(lower than normal)
10	CHWS-DP-Bias-P	E3(lower than normal); E4(lower than normal); E5(lower than normal); E7.1(higher than normal); E7.2(higher than normal);

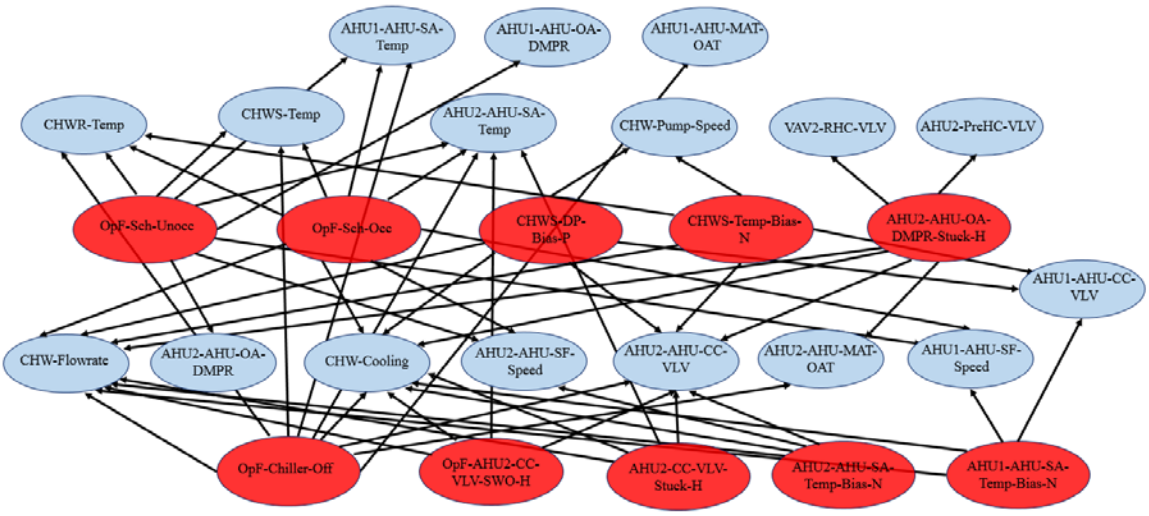


Figure 5-2 BN structure for whole building fault diagnosis

In the development of BN structure for whole buil**d**ing fault diagnosis, GeNIe and jSMILE [107] BN tools developed by the Pittsburg University are used to generate the whole building level BN.

5.4 Development of BN Parameter

The determination of parameters is the second step when developing a BN based diagnosis tool. The parameters of a BN reflect the quantitative relations among parent nodes and child nodes, expressed by using probability. Usually, three probabilities, i.e., *prior probabilities, conditional probabilities, and leak probabilities* (which is only for Noisy-Max gate, see discussion later), need to be determined when developing a BN parameter model.

Prior probability, which represents the probability of a fault event that may occur, needs to be assigned to each of the fault node.

Conditional probability is to measure the probability of a symptom event under the occurrence of a fault event. When developing a diagnostic BN, the conditional probability distributions for each evidence node are stored in conditional probability tables (CPTs) which reflect all possible combinations of states of fault nodes.

There are two methods of generating prior or conditional probabilities: from expert knowledge or from the probabilistic analysis of historical data [103]. Here the historical data can be historical operational data, experimental data, or simulated data. Data driven or machine learning techniques are typically used to obtain the probabilities when they are determined from data.

LEAK probability is the probability of the child node having a value 1 when all parent nodes are with value 0. In fault diagnosis, *LEAK* probability represents a probability of an evidence node demonstrating an abnormal (faulty) state when no fault exists.

Obtaining the prior and conditional probability distributions is the major challenges of applying a BN for FDD in a building system. Although operational data that contain naturally occurred faults are easy to obtain, ground truth data (such as a service record) that show the root causes of these faults are very hard to obtain. Even such data exists for a specific building system, the learned probabilities are not scalable to other buildings and/or building systems due to the fact that each building system is very different from others. Similar reasons prevent learning these probability distributions from simulated data.

Another challenge when developing a BN parameter model is that the number of parameters for the evidence nodes grows exponentially with the increasing number of the fault node as the network structure becomes more complex. Therefore, it is unreliable to directly generate the conditional probabilities for each state of the evidence node when there are more than four fault nodes [108]. For example, traditionally, when generating a conditional probability table of an evidence node, all of the possible combinations of states of its parent nodes (fault nodes) should be considered. If an evidence node is a Boolean state node, which contains two state, and there are n parent nodes in which Boolean state is, then the parameters of a conditional probability table would have a size of 2^{n+1}. In real practice, the development of such kind of conditions will be highly challenging and time-consuming. Therefore, canonical models such as Noisy-OR gate and Noisy-Max gate which only require a few parameters draw more attention when developing BN parameter model. The use of canonical models not only simplifies the construction of Bayesian

networks and influences diagrams, but also leads to more efficient computations [108]. In the Noisy-OR gate, effect evidence Y (a binary random variable) is deduced by each cause event X_i (a binary random variable) which acts independently of the other cause events [109]. A Noisy-OR gate model is illustrated in Figure 5-3. A Noisy-MAX gate is the extension of Noisy-OR gate. In a Noisy-MAX gate, effect event variable Y has n_y states and these states are ordered according to the effect strength. At the same time, every parent variable (cause event) X_i has n_i values. By this means, the number of parameters can be reduced from exponential to linear in the number of parent nodes.

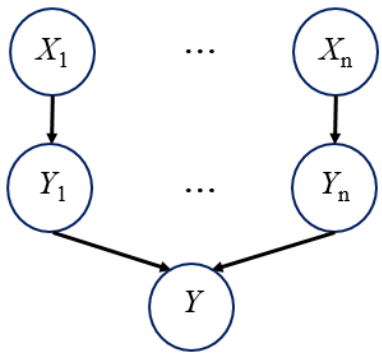


Figure 5-3 Illustration of Noisy-OR gate

Here, the Noisy-Max gate is adopted to develop the BN parameter for the evidence node. Three assumptions should be made when developing a Noisy model: (a) the child node and all its parents must be variables indicating the degree of presence of an anomaly; (b) each of the parent node must represent a cause that can produce the effect (the child node) in the absence of the other causes; and (c) there may be no significant synergies among the causes [108].

By employing canonical models, the increase of parameters in a conditional probability table is reduced to a linear increase instead of an exponent increase as in a typical BN structure. When using canonical models to map BN nodes, *LEAK* probability (as introduced earlier) needs to be determined in the Noisy-Max distribution model.

5.4.1 Development of Prior Probability Distribution

As discussed earlier, there is a lack of understanding as how often faults occur in a building system. In this study, prior probability values that have been reported for component level fault diagnosis [110] are adopted. These prior probability values can be updated when more system operation knowledge or fault data are available. In this research, the fault state is divided into faulty state and fault-free state. The initial prior probabilities for each fault node are assigned as 0.01 for faulty state and 0.99 as fault-free state as shown in Table 5-4. These numbers indicate that we believe that for each fault, there is only 1% probability for this fault to occur. Noted that we intentionally keep all prior probability values to be the same for all faults since we do not have any information to individually customize them. However, the developed BN is able to handle individual prior probability for each fault, should such information be available.

Table 5-4 Prior Probability for fault node

Fault Node State	Prior Probability
Fault	0.01
Fault free	0.99

5.4.2 Development of Conditional Probability Distribution

Due to the limitation of obtaining fault data and their root causes in a real building, obtaining condition probability from fault data is unrealistic. Obtaining accurate values for condition probabilities from expert knowledge is very difficult as well due to the fact that the same fault could behave slightly different in different buildings and under different weather conditions. However, expert knowledge could provide a range of condition probability for a fault and its associated fault evidences. Using the same example, when AHU outdoor air damper is stuck at a 100% open position in cooling mode (non-economizer), the fault symptoms including 1) mixed air temperature measurement has the same value as the outdoor air temperature measurement; and 2) AHU cooling coil valve has a position that is higher than normal (baseline) position). Both of these two symptoms are strong symptoms, i.e., they would occur whenever the fault occurs.

In this study, a fault evidence is firstly judged by whether it is a strong evidence or not. Three association levels, namely, strong evidence, medium evidence, and weak evidence, are used. If a fault evidence is a strong evidence, i.e., when a fault occurs, this evidence is most likely to occur, we consider that the condition probability of this evidence when a fault occurs is 90%, out of which, 45% is considered to have very sever fault symptom. In the example above, fault evidence node 1 (difference between mixed air and outdoor air temperatures) is a strong evidence and a 0.45 conditional probability is assigned to this node for a very severe fault symptom (very abnormal), a 0.45 conditional probability is assigned to this node for a severe fault symptom, and a 0.1 conditional probability is assigned to this node as a low severity fault symptom. A similar treatment is used for the other two association levels, i.e., medium evidence nodes and weak evidence nodes. Details are provided in Table 5-5. The conditional probabilities in Table 5-5 can be adjusted and updated when more knowledge is obtained during the system operation. In this research, the conditional probability values are based on those reported in BN based fault diagnosis tool for component-level faults[110].

Table 5-5 CPT for evidence node

Association Between Evidence Node and Fault Node	Severity	Conditional Probability Under Fault
Strong Evidence	Very Severe (S-V-S)	0.45
	Severe (S-S)	0.45
	Low Severity (S-L)	0.1
Medium Evidence	Very Severe (M-V-S)	0.25
	Severe (M-S)	0.25
	Low Severity (M-S-L)	0.5
Weak Evidence	Very Severe (W-V-S)	0.05
	Severe (W-S)	0.05
	Low Severity (W-S-L)	0.9

5.4.3 Development of *LEAK* Probability Distribution

LEAK probability represents the probability of an evidence node to be abnormal when all of the parent fault nodes are absent (when no fault occurs). The *LEAK* probability value for each evidence node is obtained by identifying the outliers in the baseline data. An outlier is defined as:

$$|x(i) - \bar{x}| > t \cdot \sigma \tag{25}$$

where $\bar{x}$ is the mean of the data sequence, σ is the standard deviation and t is the threshold.

In this study, two classes of threshold, i.e., 2σ for "very high/very low" and 1σ for "high/low" are used. σ is the standard deviation.

Therefore, *LEAK* probability can be calculated as:

$$LEAK\ Probability = \frac{Number\ of\ outlier\ data\ sample}{Total\ number\ of\ baseline\ data\ sample} \tag{26}$$

WPM method [111] is firstly used to identify baseline data that has a similar weather condition as the incoming snapshot data. The outliers in the baseline data are then counted by using the pre-defined thresholds. *LEAK* probability distribution can be obtained through Equation 26. An overall *LEAK* probability for a fault evidence is eventually calculated by averaging *LEAK* probability from the entire baseline database.

Figure 5-4 to Figure 5-7 show the outlier analysis results of the four fault symptom measurements in the chiller plant. Table 5-6 summarizes all of the *LEAK* probabilities for the BN evidence nodes based on the baseline data.

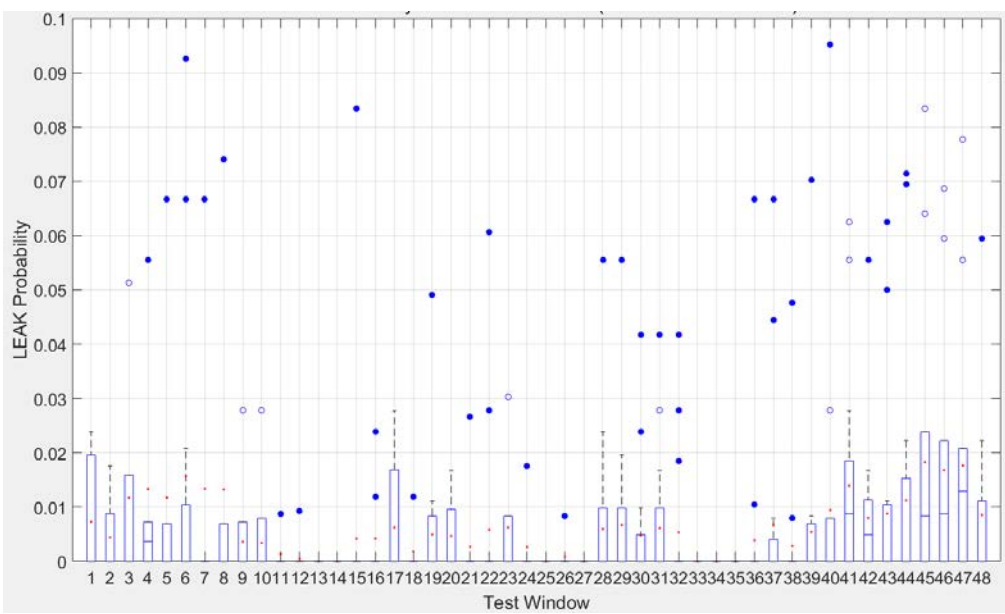


Figure 5-4 *LEAK* probability (chilled water flowrate)

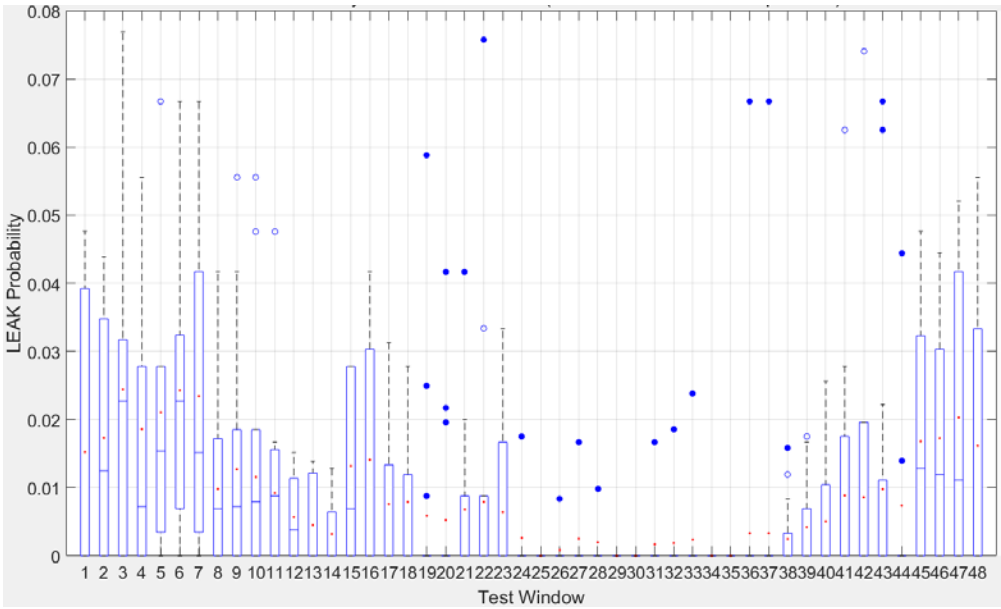


Figure 5-5 *LEAK* probability (chilled water return temperature)

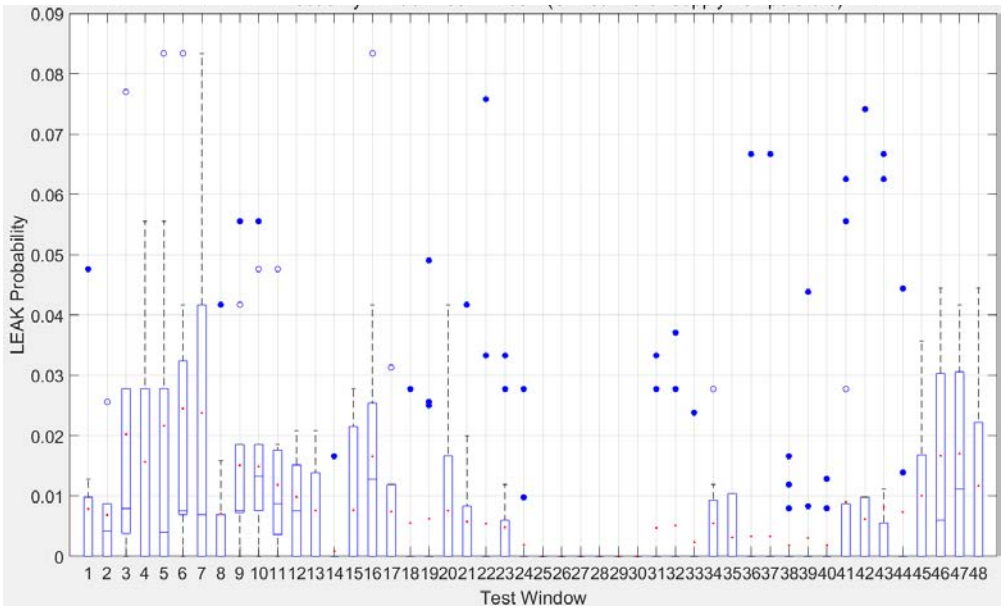


Figure 5-6 *LEAK* probability (chilled water supply temperature)

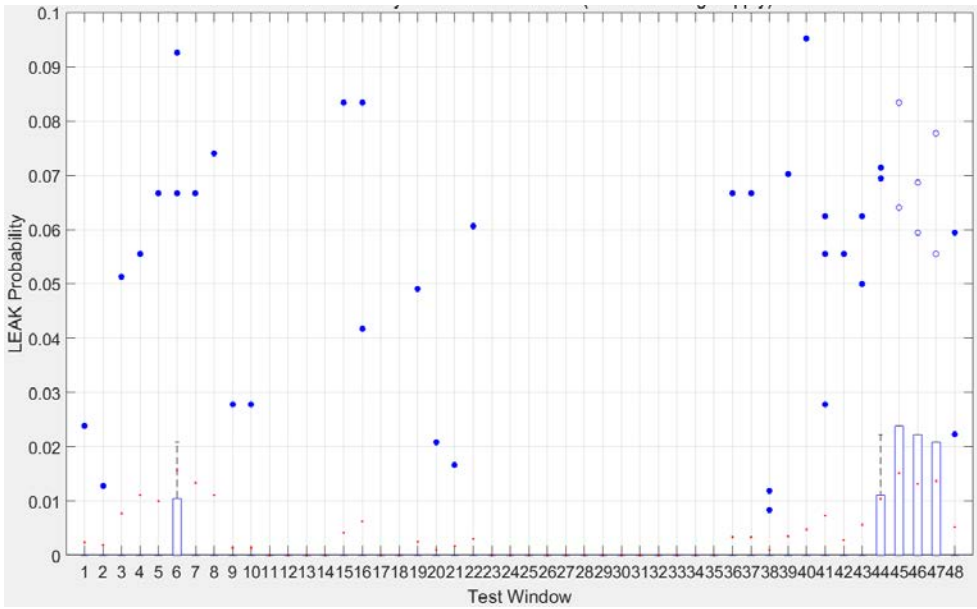


Figure 5-7 *LEAK* probability (chiller cooling energy)

Table 5-6 *LEAK* probabilities for each measurement

Fault Evidence Node	Leak Probability				Normal
	Very High	High	Very Low	Low	
CHW-Flowrate	1.0%	1.0%	1.0%	1.0%	96.0%
CHWS -Temp	1.0%	1.0%	1.0%	1.0%	96.0%
CHWR-Temp	0.5%	0.5%	0.5%	0.5%	98.0%
CHW-Cooling	0.1%	0.1%	0.1%	0.1%	99.6%
CHW-Pump-Speed	1.0%	1.0%	1.0%	1.0%	96.0%
AHU-SA-Temp	1.0%	1.0%	1.0%	1.0%	96.0%
AHU-SF-Speed	1.0%	1.0%	1.0%	1.0%	96.0%
AHU-CC-CLC	1.0%	1.0%	1.0%	1.0%	96.0%
AHU-OA-DMPR	1.0%	1.0%	1.0%	1.0%	96.0%
AHU-MAT-OAT	1.0%	1.0%	1.0%	1.0%	96.0%
VAV-RHC-VLV	1.0%	2.0%	1.0%	2.0%	94.0%

As can be observed from Table 5-6, the leak probability is generally less than 6% for the evidence nodes considered.

5.5 Fault Isolation Rule

This research, especially the evaluation process focuses on single fault scenario, i.e., the abnormality is caused by one major fault. Hence fault isolation rules are needed since there are typically a list of faults diagnosed by BN with posterior probabilities. In this study, a fault is isolated, i.e., identified as the root-cause for an abnormality, by the following two isolation rules: 1) the posterior probability of this fault node is higher than 15%; and 2) the posterior probability of this fault node is the highest among all fault nodes and is 10% higher than the second-highest one.

5.6 Summary

In this Chapter, a WPM-BN method for whole building fault diagnosis is developed. The developed WPM-BN method includes a BN structure model and associated BN parameters. Expert knowledge is used to generate the BN rules. The BN structure model includes a two-layer configuration including a fault layer and a fault evidence layer. In the fault layer, the identified root cause faults are assigned to the fault nodes. In the fault evidence layer, sensor measurements are mainly included to generate fault evidence nodes. BN parameters including prior probability distribution and conditional probability distribution are determined by reported literature values and employing expert knowledge. *LEAK* probability distribution is learned through the WPM baseline dataset in an automated manner. The evaluation of this method is discussed in Chapter 6.

Chapter 6 Method Evaluation

6.1 Ground Truth Establishment

6.1.1 Introduction

As described in Chapter 2, the Nesbitt Hall at Drexel University is used as the demonstration building in this project. BAS data from the demonstration building from June 1st 2016 to September 31st 2018 are collected. During this time period, there are about thirty-four (34) days in which artificial faults have been implemented (a total of twenty-seven (27) days between 2016 to 2017, and 7 days in 2018, as described in Chapter 8) and three (3) additional days in which natural faults have occurred. In order to evaluate the developed AFDD methods, the collected data need to be manually analyzed and labelled, i.e., to establish a "ground truth". Noted that there are several scenarios that could happen in a day:

- Scenario 1 – an artificial fault is implemented and expected fault symptoms are observed. This test day will be labelled as a fault test day or fault test case.
- Scenario 2 – no artificial fault is implemented but naturally occurred faults have been observed which have caused the whole building performance to be different from its baseline (details in Section 6.1.2). This test day will be labelled as a fault test day or fault test case.
- Scenario 3 – an artificial fault is implemented and natural faults have occurred simultaneously. The whole building performance is different from its baseline (details in Section 6.1.2) but due to the complications from naturally occurred faults, the artificially implemented fault has not yielded expected symptoms. This test day will be labelled as a fault test day or fault test case, although the faults are more than the artificially implemented fault.
- Scenario 4 – an artificial fault is implemented but no expected fault symptom is observed. The building performance is similar to its baseline (details in Section 6.1.2). This test day will be labelled as a fault free test day or fault free test case.

Therefore, this study uses fault symptoms to differentiate a fault test case from a fault free test case. Although in a fault free test case, faults (artificially implemented or natural faults) could still exist in the system. However, these faults do not cause the building performance to be different from its baseline.

6.1.2 Method of Ground Truth Establishment

A thorough manual examination is firstly performed to identify and tag collected building data into two states, i.e., "fault free" and "faulty" as described above. Whole building operation is considered as "faulty", when there are observable abnormalities found in more than two key BAS measurements from two different subsystems respectively (listed in Table 6-1). A key measurement is considered to be abnormal if the difference between its value and its baseline value under similar weather conditions, is larger than a threshold. The abnormality thresholds are summarized in Table 6-1. Notice that these thresholds are not used in our AFDD tools but are used when we manually tag the test data for setting a "ground truth". These thresholds are selected based on 1) sensor accuracy; and 2) measurement noise and fluctuation under normal operation.

Table 6-1 Key BAS measurements and abnormality threshold

Subsystem	Key Measurement	Abnormality Threshold (compared with baseline)
Primary cooling subsystem (Chiller plant)	1) Chilled water supply temperature (°F)	± 2 °F
	2) Differential chilled water pressure (Psi)	± 0.5 Psi
	3) Chilled water supply pump speed (%)	± 5%
	4) Chilled water return temperature (°F)	± 5 °F
	5) Chilled water flowrate (gpm)	± 50 gpm
Supply air subsystem (AHUs)	1) Supply air temperature (°F)	± 3 °F
	2) Supply air differential air pressure (inch of water.)	± 0.3 inch of water
	3) Supply air fan speed (%)	± 5%
	4) Cooling coil valve open position (%)	± 15%
	5) Outdoor air damper open position (%)	± 10%

For each fault test case, the root cause for the abnormality is again manually identified by observing these key BAS measurements and their baseline values. This manual process is necessary because 1) faults could naturally occur during the test days when we did not implement a fault; 2) even a fault was artificially implemented, it might not cause any abnormality under certain operational conditions. Such data is again labeled as "fault free" for this project. And 3) even a fault was artificially implemented, other naturally occurred faults could still occurred at the same time, and caused abnormality.

6.1.3 Whole Building Fault Identification and Fault Free Day Generation

Out of the twenty-seven (27) fault test days, in which faults have been artificially implemented, eleven (11) test cases have been labeled as "fault test case". Two (2) fault test days (August 13th, 2016 and August 23rd, 2017) are identified as failure cases because no enough data samples are collected due to a BAS failure. Furthermore, three naturally occurred faults (in July 06th, 2016, January 14th and July 09th, 2017, respectively) have been identified to have whole build level impacts. Hence there are in total of fourteen (14) fault test cases used as the ground truth as listed in Table 6-2. From the 450 available days, we have selected fourteen (14) fault-free test cases from different seasons for the later false alarm evaluation: ten (10) days in summer season, two (2) days in transitional season, and two (2) days in winter season.

6.2 Evaluation of Fault Detection

6.2.1 Fault Detection

A total of fourteen (14) whole building fault test cases (including three (3) naturally occurred whole building fault test cases) are used to evaluate the fault detection result. For method evaluation purpose, two PC retention rates, i.e., 0.65 and 0.95, are used to evaluate the fault detection results respectively. Usually, a higher PCs retention rates will increase the sensitivity of the method and increase the fault detection rate (i.e., it is easier to detect faults) because more cumulative variance can be captured by increase the percentage of PC, but it will also have a risk of increasing the false alarm rate. Under PC retention rate of 0.65, a total of eleven (11) fault test

cases have been successfully detected. Detection rate is hence 79%. Under the PC retention rate of 0.95, all fourteen (14) fault test cases have been successfully detected. Detection rate is hence 100%.

Table 6-3 lists the fault detection results.

6.2.2 Successful Detection Case Examples

6.2.2.1 AHU Outdoor Air Damper Stuck Fault at a Higher than Normal Position Fault

1. Fault Description

On July 11th, 2017, a damper stuck fault (stuck at a higher than normal position) was implemented at AHU-2. The outdoor air damper at AHU-2 was artificially stuck at 90% and 100% open positions by overriding the corresponding control signal in the BAS from 10:00AM to 08:01PM. The stuck positions (90% open) was higher than the damper's normal position (15% open) under similar weather conditions. In the summer season when HVAC operates under cooling mode, the AHU outdoor air damper is usually controlled to be at 15% open to maintain the minimum requirement of fresh air flowrate. Under such circumstances, if the outdoor air damper is stuck at a position that is higher than normal (15% during cooling mode), cooling coil valve position will be increased to ensure the supply air temperature meet the setpoint requirement. However, in the early morning and evening hours, when outdoor enthalpy is lower than the return air enthalpy, the outdoor air damper is controlled under economizer mode to save energy. In these situations, the damper stuck fault would not yield a strong fault impact, as the stuck position is very similar to the normal damper position (under economizer mode).

2. Fault Detection Result

Figure 6-1 shows the fault detection result. It can be see that both of the faults are successfully detected by the WPM-FPCA method. However, in some time periods (especially early morning and evening hours), T^2 statistic value does not overpass the threshold due to the lack of fault impacts.

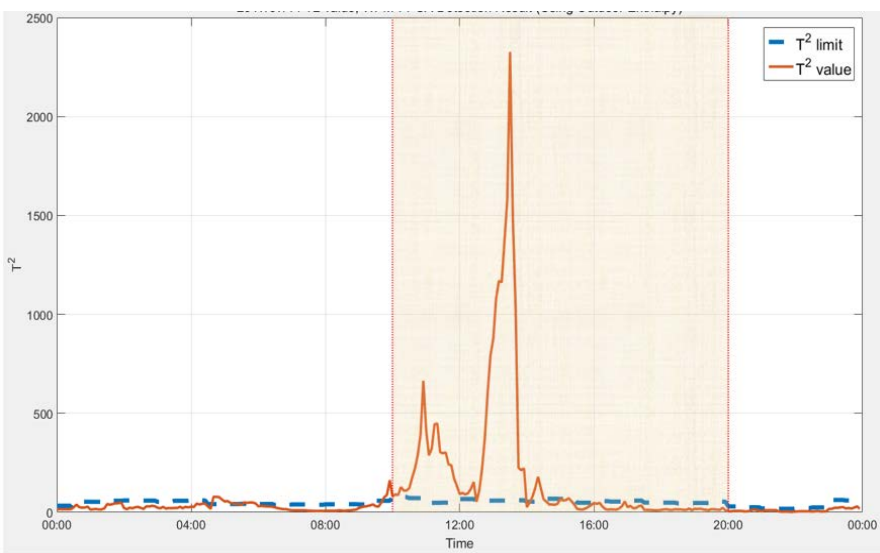


Figure 6-1 Fault detection result under 0.95 PC retention rate (July 11th, 2017)

Table 6-2 Whole building fault identification (Cases from 2016 to 2017)

Fault Type	Date	Whole Building Fault Description	Whole Building Fault Identification	Equipment Analysis			
				Chiller	AHU-1	AHU-2	AHU-3
OpF-Sch-Unocc	07/06/16	System stopped at 4:00PM to 23:30PM	Whole Building Fault (naturally occurred)	Abnormality	Abnormality	Abnormality	Abnormality
OpF-Chiller-Off	09/11/16	Operator fault, chiller off	Whole Building Fault	Abnormality	Abnormality	Abnormality	Abnormality
	07/09/17	Chiller stopped earlier	Whole Building Fault (naturally occurred)	Abnormality	Abnormality	Abnormality	Abnormality
OpF-Sch-Occ	01/14/17	Occupied from 1:30 AM to 7:00 AM	Whole Building Fault (naturally occurred)	Abnormality	Abnormality	Abnormality	Abnormality
	08/05/17	HVAC system earlier occupied	Whole Building Fault	Abnormality	Abnormality	Abnormality	Abnormality
OpF-CC-VLV-SWO-H	08/11/17	AHU-2 cooling coil valve position software override fault (higher than normal)	Whole Building Fault	Abnormality	Not observed	Abnormality	Not observed
CHWS-DP-Bias-P	09/15/17	Chiller chilled water differential pressure sensor positive bias 0.1 psi	Whole Building Fault	Abnormality	Abnormality	Abnormality	Abnormality
CHWS-Temp-Bias-N	08/03/17	Chiller chilled water supply temperature sensor negative bias 4°F	Whole Building Fault	Abnormality	Abnormality	Not observed	Not observed
AHU-SA-Temp-Bias-N	08/08/16	AHU-1 supply air temperature sensor negative bias 4°F	Whole Building Fault	Abnormality	Abnormality	Not observed	Not observed
	09/07/16	AHU-2 supply air temperature sensor negative bias 4°F	Whole Building Fault	Abnormality	Not observed	Abnormality	Not observed
AHU-OA-DMPR-Stuck-H	12/01/16	AHU-2 outdoor air damper stuck at 90% open (higher than normal)	Whole Building Fault	Abnormality	Not observed	Abnormality	Not observed
	01/03/17	AHU-2 outdoor air damper stuck at 80% open (higher than normal)	Whole Building Fault	Abnormality	Not observed	Abnormality	Not observed
	07/11/17	AHU-2 outdoor air damper stuck at 90% open (higher than normal)	Whole Building Fault	Abnormality	Not observed	Abnormality	Not observed
	07/18/17	AHU-2 outdoor air damper stuck at 100% open (higher than normal)	Whole Building Fault	Abnormality	Not observed	Abnormality	Not observed

Table 6-3 Fault detection result (Cases from 2016 to 2017)

Fault Type	Date	Whole Building Fault Description	Implemented Fault Comments	Fault Detection Result (under PC 0.65)	Fault Detection Result (under PC 0.95)
OpF-Sch-Unocc	07/06/16	System stopped at 4:00PM to 23:30PM	Whole Building Fault (naturally occurred)	Detected	Detected
OpF-Chiller-Off	09/11/16	Operator fault, chiller off	Whole Building Fault	Detected	Detected
	07/09/17	Chiller stopped earlier	Whole Building Fault (naturally occurred)	Detected	Detected
OpF-Sch-Occ	01/14/17	Occupied from 1:30 AM to 7:00 AM	Whole Building Fault (naturally occurred)	Detected	Detected
	08/05/17	HVAC system earlier occupied	Whole Building Fault	Detected	Detected
OpF-CC-VLV-SWO-H	08/11/17	AHU-2 cooling coil valve position software override fault (higher than normal)	Whole Building Fault	Detected	Detected
CHWS-DP-Bias-P	09/15/17	Chiller chilled water differential pressure sensor positive bias 0.1 psi	Whole Building Fault	Detected	Detected
CHWS-Temp-Bias-N	08/03/17	Chiller chilled water supply temperature sensor negative bias 4°F	Whole Building Fault	Mis-detected	Detected
AHU-SA-Temp-Bias-N	08/08/16	AHU-1 supply air temperature sensor negative bias 4°F	Whole Building Fault	Mis-detected	Detected
	09/07/16	AHU-2 supply air temperature sensor negative bias 4°F	Whole Building Fault	Detected	Detected
AHU-OA-DMPR-Stuck-H	12/01/16	AHU-2 outdoor air damper stuck at 90% open (higher than normal)	Whole Building Fault	Detected	Detected
	01/03/17	AHU-2 outdoor air damper stuck at 80% open (higher than normal)	Whole Building Fault	Detected	Detected
	07/11/17	AHU-2 outdoor air damper stuck at 90% open (higher than normal)	Whole Building Fault	Detected	Detected
	07/18/17	AHU-2 outdoor air damper stuck at 100% open (higher than normal)	Whole Building Fault	Mis-detected (Only one period detected)	Detected

6.2.2.2 Chiller Stop Fault (Naturally Occurred Fault)

1. Fault Description

On July 9th, 2017, the chiller was stopped from 4:00-15:30 abnormally as shown in Figure 6-2. This is a naturally occurred because according to the system scheduling, the chiller plant should be operated from 4:00 to 21:00. Meanwhile, between 15:30 to 20:00 in that day, the secondary systems also have naturally occurred abnormalities.

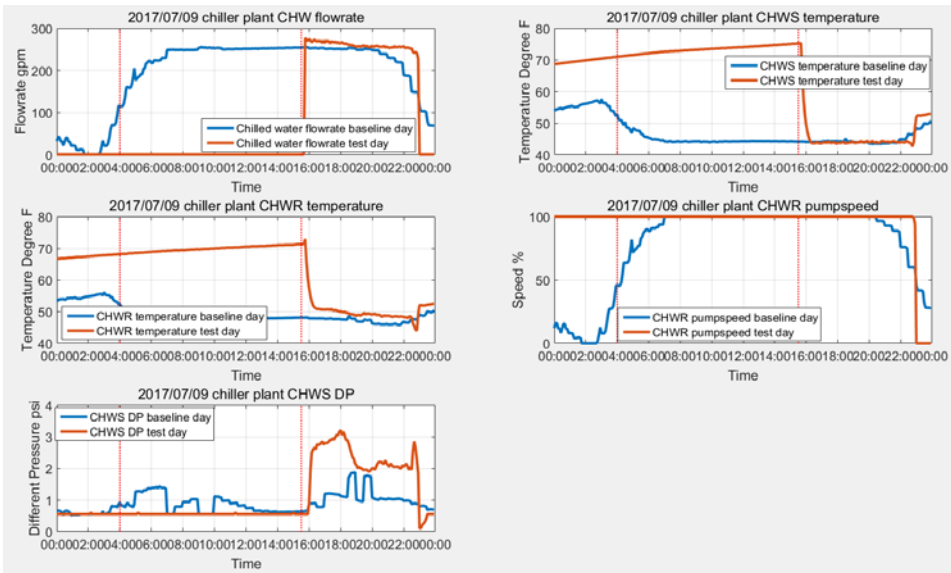


Figure 6-2 Chiller plant operation status (July 9th, 2017)

2. Fault Detection Result

Figure 6-3 shows the fault detection result. It can be seen that the T^2 statistic value overpasses the threshold tremendously in two periods (4:00-15:30 for period I and 15:30 to 20:00 for period II). For the first period (4:00-15:30), this is due to the chiller stop fault. Therefore, the fault detection method successfully detected this whole building fault.

For the second period, the fault detection method also reports that there exists whole building level abnormality, although the chiller was operated normally. An investigation on the downstream subsystems illustrates that there are operation abnormalities in multi-subsystems. For example, the cooling coil valve position in all AHU-1 and AHU-2 did not return to their normal positions compared with the baseline value until 20:00 as can be seen in Figure 6-4. The reason for these abnormalities are believed to be caused by the thermal storage of the building. Due to the chiller stop fault, the building was over-heated during the day. Hence the secondary systems, after the chiller returned to be normal, were operated very differently from their baseline status. An example from the VAV-1-1-3 and VAV-2-3-3 clearly demonstrate this operation adjustment as show in Figure 6-5. In this figure, it can be seen that the zone temperature begins to decrease in 15:30 when the chiller started to operate, but only reach the normal level around 20:00.

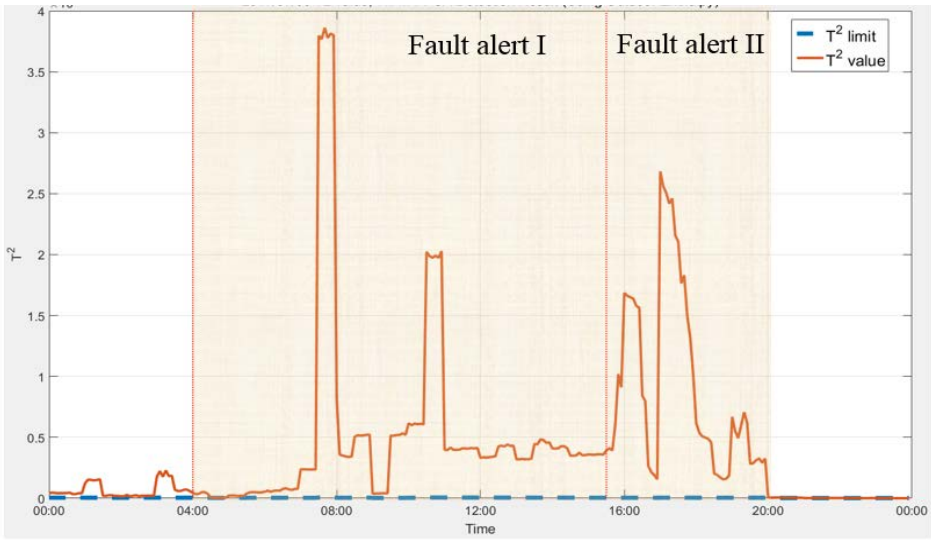


Figure 6-3 Fault detection result under 0.95 PC retention rate (July 9th, 2017)

Hence, it is concluded that the WPM-FPCA has reported correctly that there exists a whole building level abnormality, although it is an aftermath from a previous fault.

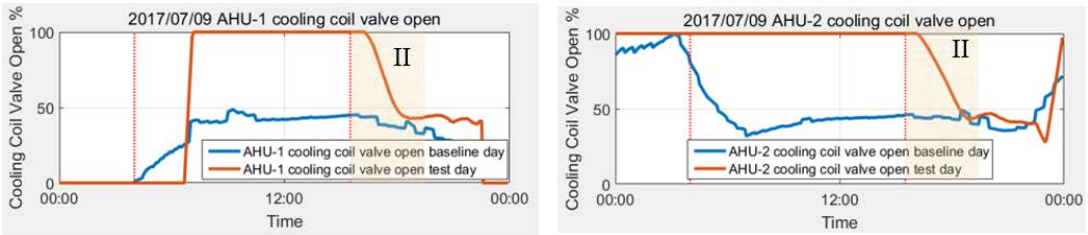


Figure 6-4 AHU-1 and AHU-2 cooling coil operation status (July 9th, 2017)

6.2.3 Mis-detection Case Example

Some faults were not detected under 0.65 PC retention rate because even if there were symptoms happening in different sub-systems, these symptoms were not too many, so under lower 0.65 PC retention rate, the method is not sensitive to these fault. But when PC retention rate is increased, these minor variances in different measurements can easily detected. A fault test case from August 3th, 2017 is illustrated as below.

1. Fault Description

On August 3th, 2017, a chilled water supply temperature sensor negative bias of 4 °F fault was implemented by adjusting the chilled water outlet temperature setpoint on the chiller control panel. The fault test period was from 10:00 to 21:27. This fault had impact on chilled water return temperature (as shown in Figure 6-6) and also had impacts on the downstream subsystems. During

the test, we observed that the cooling coil valves in AHU-1 had higher than normal positions to allow larger chilled water flowrate to meet the cooling load as shown in Figure 6-7.

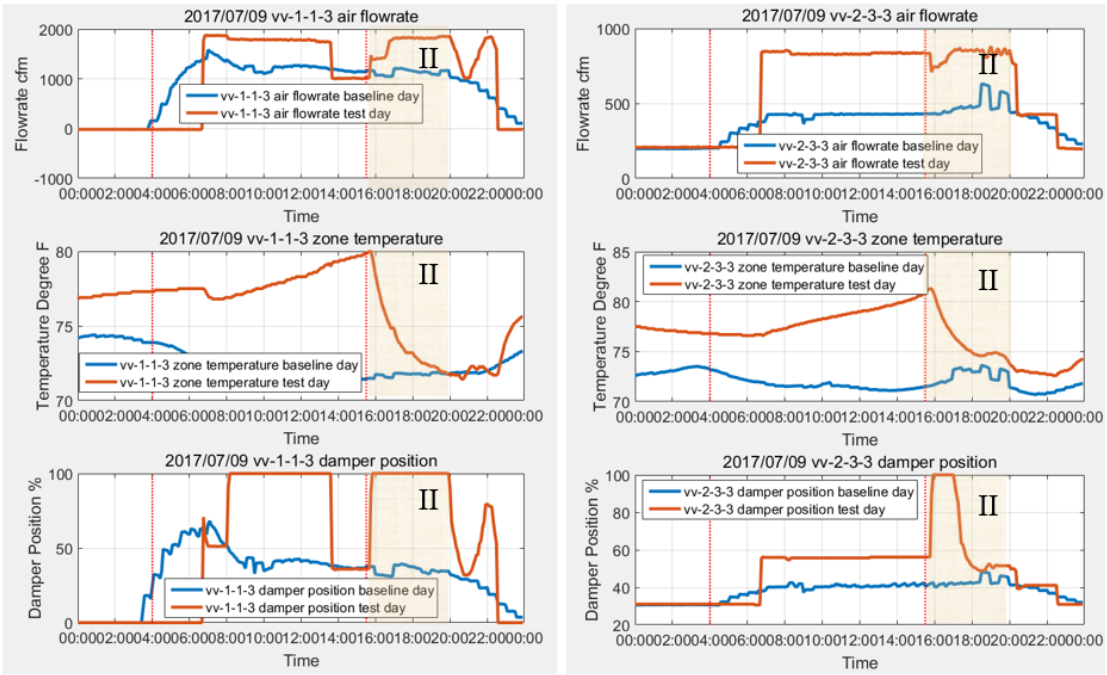


Figure 6-5 VAV terminal unit operation status (July 9th, 2017)

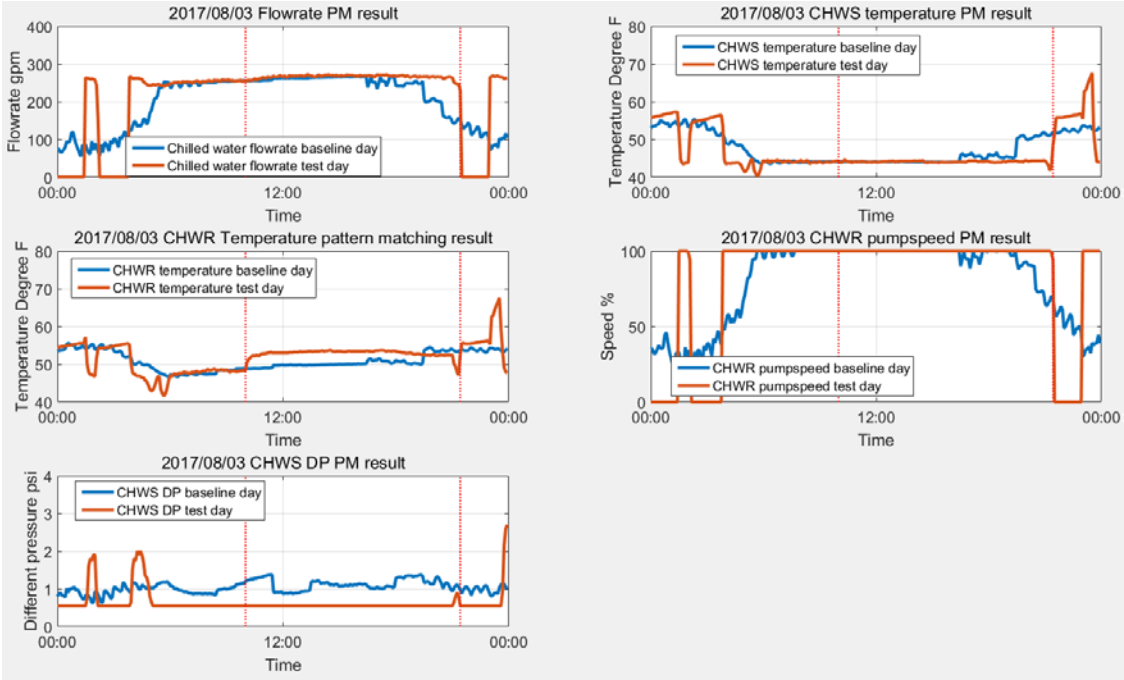


Figure 6-6 Chiller operation status (August 3rd, 2017)

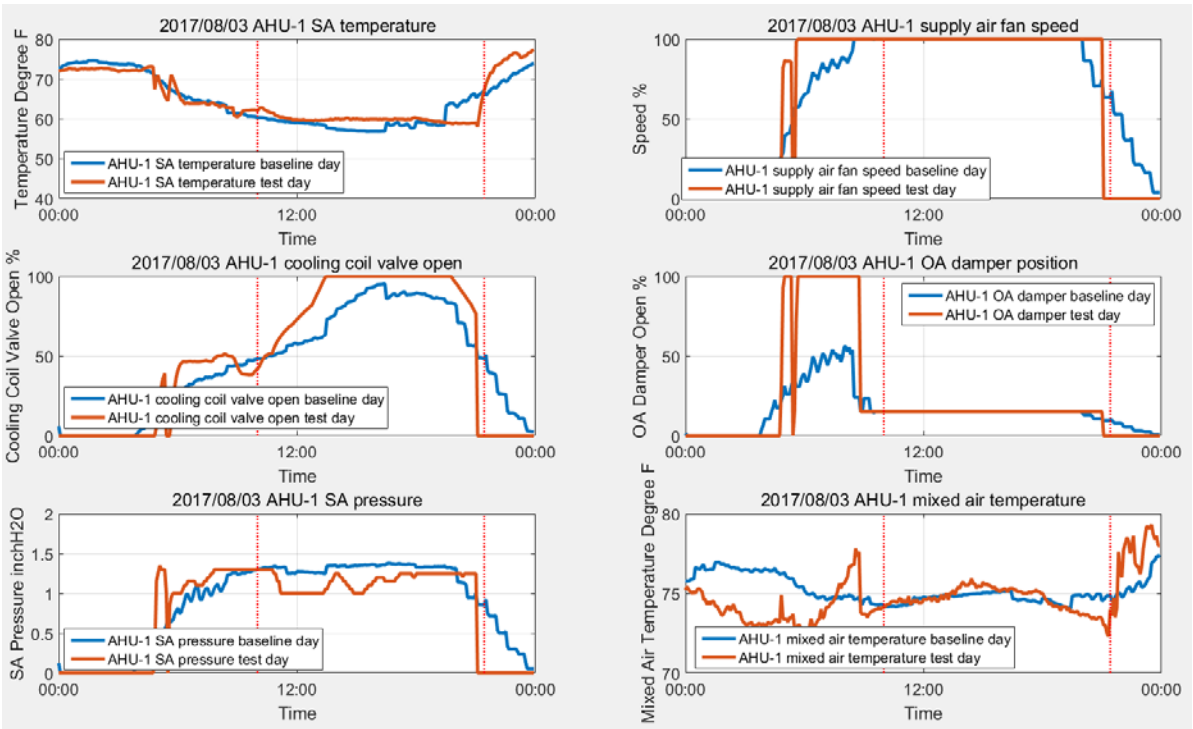


Figure 6-7 AHU-1 operation status (July 9th, 2017)

2. Fault Detection Result

This fault is not detected by the WPM-FPCA method under 0.65 PC retention rate as shown in Figure 6-8. But when increasing the PC retention rate to 0.95, this fault is successfully detected as can be seen in Figure 6-9. In this research, we set PC retention rate at 0.65 which is relatively conservative, so in this case, we treat it as mis-detected case.

6.2.4 Evaluation on False Alarm Rate

Although it is desired for automated fault detection tools to have a high detection accuracy, low false alarm rate is another criteria to assess the fault detection strategy. Typically, a high false alarm rate may lead to the building operators to ignore or even disable the FDD tool and would decrease the system performance. A total of fourteen (14) fault free test cases are randomly selected from the four hundred and fifty (450) available days, including ten (10) fault free test cases in the summer season, two (2) fault free test cases in the transition season and two (2) fault free test cases in the winter season in 2016 and 2017. These fault free test cases are used to evaluate the false alarm rate for the WPM-FPCA fault detection method under two PC retention rates as shown in Table 6-4. It can be observed that no false alarm has been reported.

Three fault free days from summer season, transition season and winter season, i.e., August 28th 2016, December 8th 2016 and January 24th 2017 are used here to illustrate the typical detection results as shown from Figure 6-10 to Figure 6-12. It can be seen that the T^2 of the fault data are below their baseline thresholds for three days respectively.

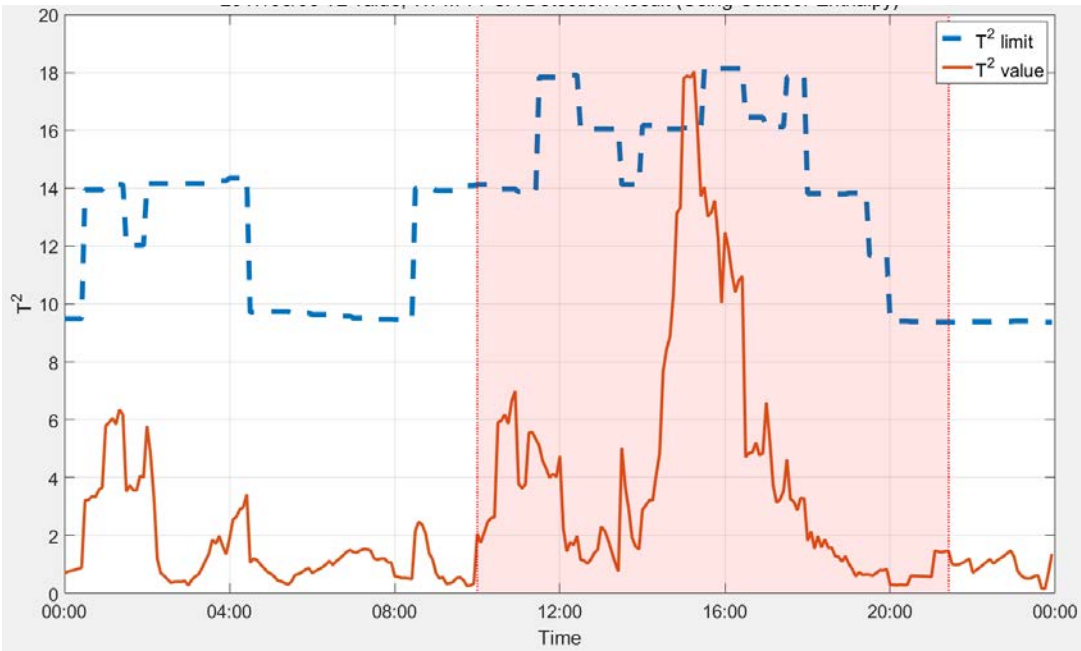


Figure 6-8 Fault detection result (August 3rd, 2017) under 0.65 PC retention rate

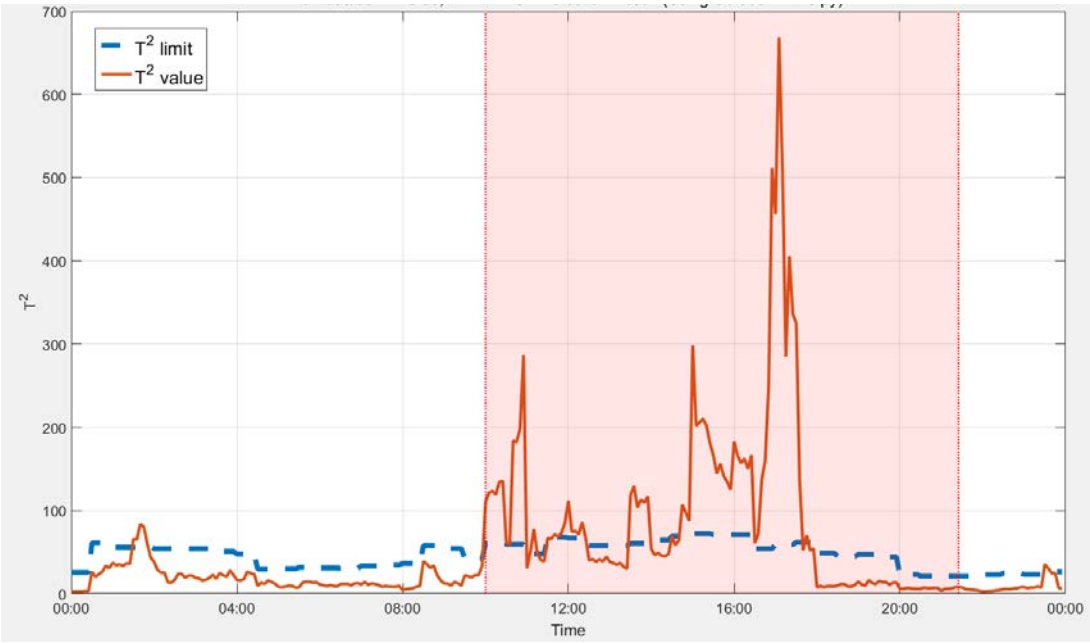


Figure 6-9 Fault detection result (August 3rd, 2017) under 0.95 PC retention rate

Table 6-4 False alarm analysis

Test Date	Season	Detection Result (0.65 PC retention rate)	Detection Result (0.95 PC retention rate)
2016.07.26	Summer	Fault free	Fault free
2016.08.26	Summer	Fault free	Fault free
2016.08.27	Summer	Fault free	Fault free
2016.08.28	Summer	Fault free	Fault free
2016.08.29	Summer	Fault free	Fault free
2017.07.02	Summer	Fault free	Fault free
2017.07.05	Summer	Fault free	Fault free
2017.08.06	Summer	Fault free	Fault free
2017.08.10	Summer	Fault free	Fault free
2017.08.12	Summer	Fault free	Fault free
2016.11.01	Transition	Fault free	Fault free
2016.12.08	Transition	Fault free	Fault free
2017.01.20	Winter	Fault free	Fault free
2017.01.24	Winter	Fault free	Fault free

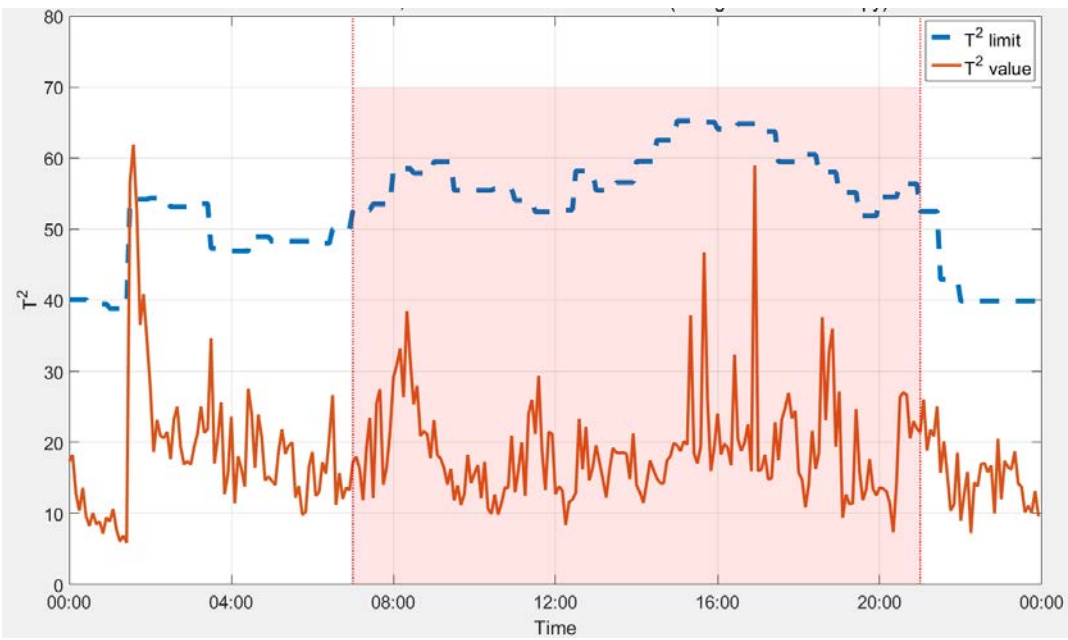


Figure 6-10 Detection result in August 28th, 2016 (summer season)

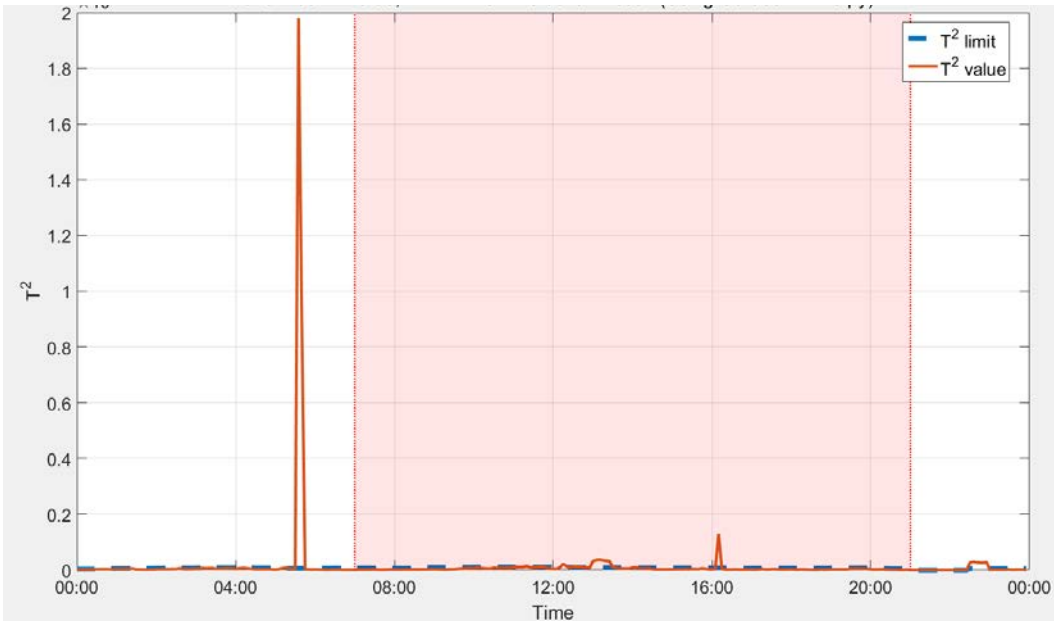


Figure 6-11 Detection result in Dec. 8th, 2016 (transition season)

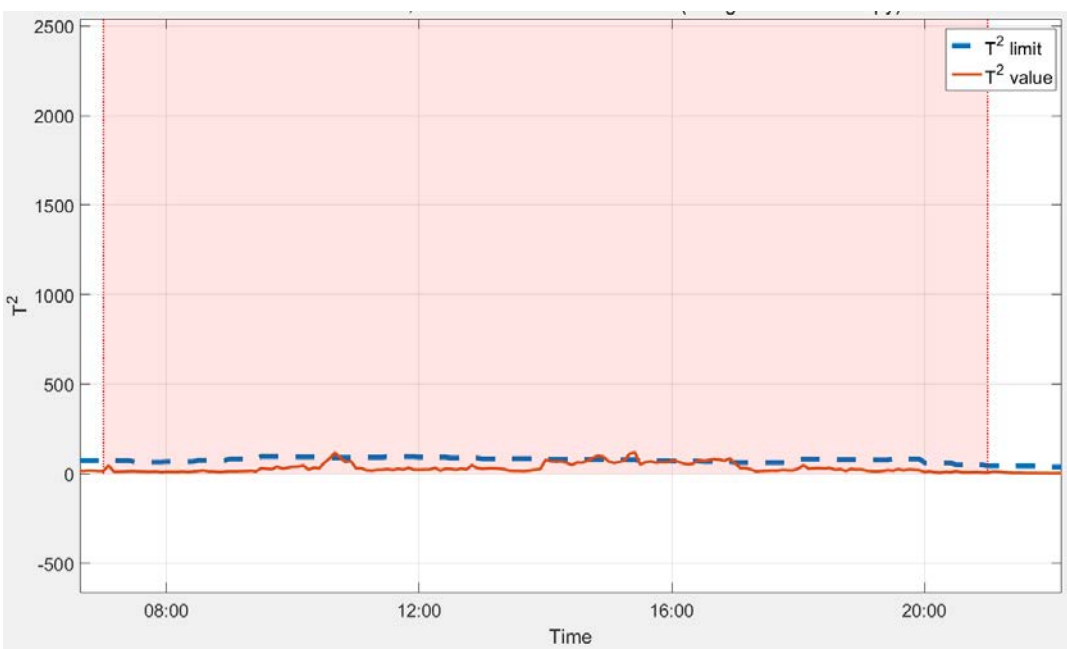


Figure 6-12 Detection result in Jan. 24th, 2017 (winter season)

6.3 Evaluation of Diagnosis Method

The fourteen (14) whole building fault test cases (including 3 naturally occurred whole building fault cases) are used to evaluate the WPM-BN fault diagnosis method described in

Chapter 5. Thirteen (13) out of fourteen (14) fault cases are successfully diagnosed by the WPM-BN method as summarized in Table 6-5. The following sections illustrate a typical successful case and also discuss the mis-diagnosed case.

6.3.1 Successful Diagnosis Case Example

1. Fault Description

Here, we use AHU Outdoor Air Damper Stuck Fault at a Higher than Normal Position Fault as illustrated in Section 6.2.2.1 to demonstrate the fault diagnosis result.

2. Fault Diagnosis Result

For the July 11th case, the method reports that there is (on average) 40% posterior probability for the "AHU-2 outdoor air damper stuck at too high fault". This posterior probability ranks No.1 compared with the posterior probabilities of other causes as shown in Figure 6-13. Therefore, this fault root-cause is successfully diagnosed.

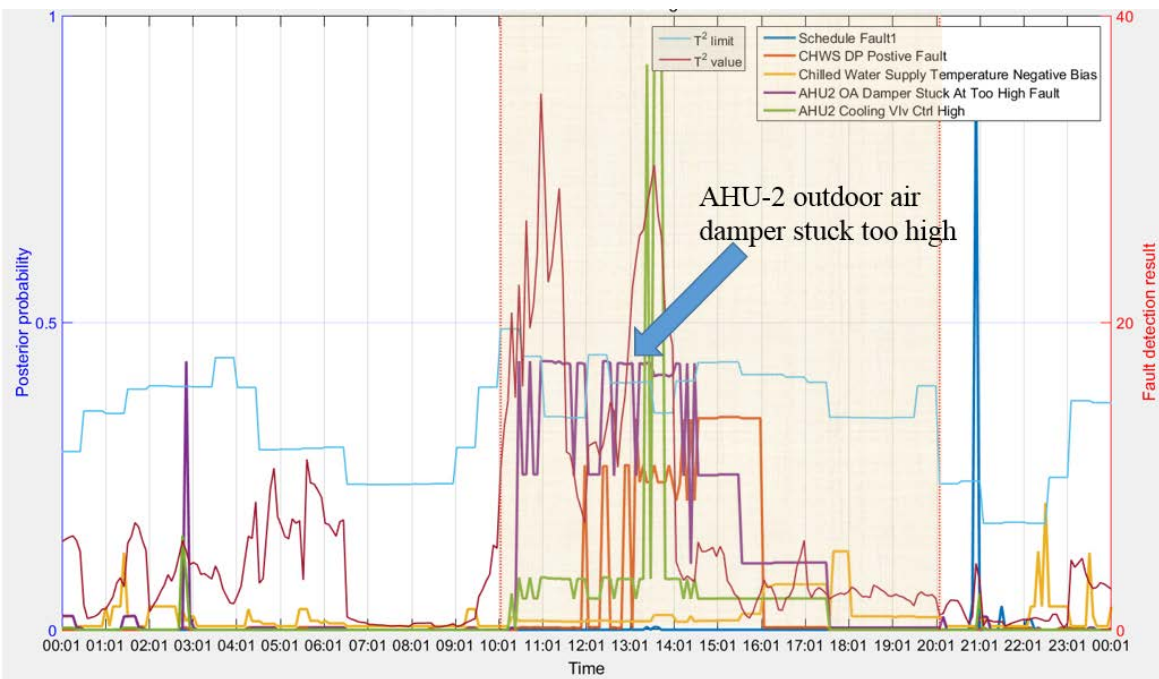


Figure 6-13 Fault detection and diagnosis result (July 11th, 2017)

Table 6-5 Fault diagnosis result (Cases from 2016 to 2017)

Fault Type	Date	Whole Building Fault Description	Implemented Fault Comments	Fault Diagnosis Result
OpF-Sch-Unocc	07/06/16	System stopped at 4:00PM to 23:30PM	Whole Building Fault (naturally occurred)	Diagnosed
OpF-Chiller-Off	09/11/16	Operator fault, chiller off	Whole Building Fault	Diagnosed
	07/09/17	Chiller stopped earlier	Whole Building Fault (naturally occurred)	Diagnosed
OpF-Sch-Occ	01/14/17	Occupied from 1:30 AM to 7:00 AM	Whole Building Fault (naturally occurred)	Diagnosed
	08/05/17	HVAC system earlier occupied	Whole Building Fault	Diagnosed
OpF-CC-VLV-SWO-H	08/11/17	AHU-2 cooling coil valve position software override fault (higher than normal)	Whole Building Fault	Diagnosed
CHWS-DP-Bias-P	09/15/17	Chiller chilled water differential pressure sensor positive bias 0.1 psi	Whole Building Fault	Diagnosed
CHWS-Temp-Bias-N	08/03/17	Chiller chilled water supply temperature sensor negative bias 4°F	Whole Building Fault	Mis-diagnosed because not too many evidences found in chiller and AHUs
AHU-SA-Temp-Bias-N	08/08/16	AHU-1 supply air temperature sensor negative bias 4°F	Whole Building Fault	Diagnosed
	09/07/16	AHU-2 supply air temperature sensor negative bias 4°F	Whole Building Fault	Diagnosed
AHU-OA-DMPR-Stuck-H	12/01/16	AHU-2 outdoor air damper stuck at 90% open (higher than normal)	Whole Building Fault	Diagnosed
	01/03/17	AHU-2 outdoor air damper stuck at 80% open (higher than normal)	Whole Building Fault	Diagnosed
	07/11/17	AHU-2 outdoor air damper stuck at 90% open (higher than normal)	Whole Building Fault	Diagnosed
	07/18/17	AHU-2 outdoor air damper stuck at 100% open (higher than normal)	Whole Building Fault	Diagnosed

6.3.2 Mis-Diagnosis Case Example

1. Fault Description

On August 3rd, 2017, a chilled water supply temperature sensor negative bias of 4 °F fault was implemented by adjusting the chilled water outlet temperature setpoint on the chiller control panel. The fault test period was from 10:00 to 21:27. This fault had impacts on the downstream subsystems. During the test, we observed that only cooling coil valves in AHU-1 had higher than normal positions and chilled water return temperature had higher than baseline value.

2. Fault Diagnosis Result

In this case, the root cause of fault was not successfully diagnosed by the WPM-BN method. In the diagnosis result, posterior probabilities for "AHU-2 cooling coil valve open too high" fault and "AHU-2 outdoor air damper stuck at too high" fault reach higher levels compared with "chilled water supply temperature negative bias" fault. The posterior probability for these fault cause are: 0.91 for "AHU-2 cooling coil valve operate at too high" fault, 0.39 for "AHU2 outdoor air damper stuck at too high position" fault, posterior probability, and 0.04 for "Chilled water temperature negative bias" fault as shown in Figure 6-14.

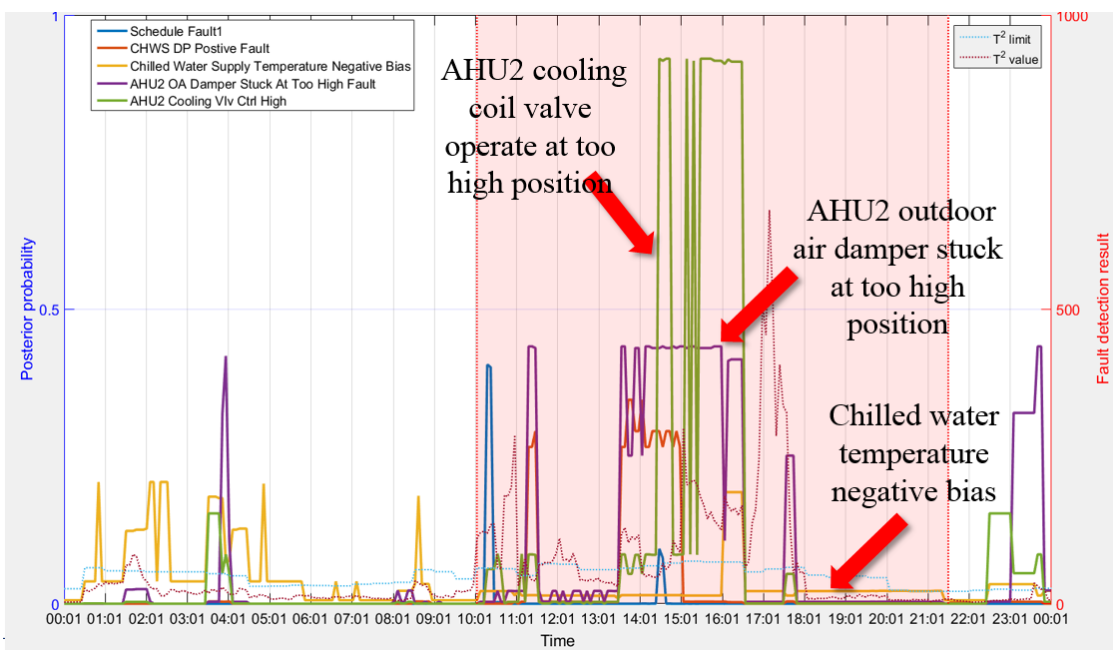


Figure 6-14 Fault diagnosis result (August 3rd , 2017)

We are currently analyzing the diagnosis results, attempting to find the reasons for this mis-diagnosis. From the analyses, one reason is that not too many evidences are found in chiller, AHUs. In this case, only chilled water return temperature and AHU-1 cooling coil valve position are found to be abnormal compared with the baseline data. As more evidences can strengthen BN inference

and hence generate a higher posterior probability, it is therefore WPM-BN fault diagnosis method does not diagnose the fault. Another potential reason could be that the conditional probability for each root fault cause is set to be equal when these faults have the same evidences. Since the "chilled water temperature negative bias" fault would cause similar fault evidences as other faults such as "cooling coil valve open too high" and "outdoor air damper stuck at too high" faults, without additional conditional probability information, the WPM-BN method would not be able to differentiate these faults that have very similar fault symptoms.

6.4 Evaluation on Method Implementation Cost and Payback

To estimate deployment costs, two metrics are investigated in this study: the number of hours required for our team to deploy the solution, and the typical costs presently associated with deploying manual retro-commissioning by service providers. Deployment costs are a function of the quantity of equipment and the quantity of data points collected from a given building. For a building that has a somewhat typical HVAC configuration, i.e., central chiller system with VAV distribution system, this value can be estimated in terms of cost per square foot. Based on conversations with many of our industrial partners, a standard estimate for a service provider to perform retro-commissioning analysis is $0.20/square foot, exclusive of performing any repairs. Comparing the time-savings that can be realized by deploying a more automated strategy, and by studying the time required to deploy our solution, it is found that utilizing this AFDD technology can save approximately 40% of this cost by removing the requirements for site-specific algorithm customization and threshold customization. As a result, the deployment cost is estimated at $0.12 per square foot.

Energy impacts of the implemented faults are also analyzed. Since weather/schedule- similar baseline dataset is generated as part of the FDD process in our tools, energy impacts are fairly easy to be obtained by comparing energy consumption measurements (real or calculated) between snapshot data and WPM baseline dataset. The detailed impact data are provided in Appendix V. Based on the expected energy impacts, the simple payback of the developed method is less than 3 years.

6.5 Summary

In this Chapter, WPM-FPCA fault detection and WPM-BN fault diagnosis tools are evaluated using the collected evaluation data set from the Nesbitt Hall. Firstly, the ground truth is established to identify fault and fault free cases using an extensive manual process. Root cause of a fault test case is also isolated. A total of eleven (11) fault test cases that contain artificially implemented whole building faults are identified from twenty-five (25) test cases. Three (3) naturally occurred faults which demonstrate whole building fault symptoms are also identified. These fourteen (14) fault test cases are used to evaluate the developed methods. The WPM-FPCA fault detection method is evaluated under different PC retention rates. The evaluation result shows that using a 0.65 PC retention rate, the developed method can reach 79% fault detection rate, and using a 0.95 PC retention rate, the developed method can reach 100% fault detection rate. Fourteen (14) fault

free test cases are randomly selected from multiple seasons. The developed WPM-FPCA method is demonstrated to have a 0% false alarm rates under both PC retention rates. In the real practice, PC retention rate can be adjusted by the user based on a building's specific requirement to balance the fault detection rate and false alarm rate. Evaluation on the WPM-BN based fault diagnosis method demonstrates that the developed method has successfully diagnosed thirteen (13) fault test cases. The one mis-diagnosed case exhibits few fault symptoms in subsystems. Overall speaking, the developed whole building AFDD tools demonstrate satisfactory detection and diagnosis results with low false alarm rate. Furthermore, the deployment cost of the developed whole building AFDD tools is expected to be at least 40% lower than other market available AFDD tools, due to the data-driven and highly automated nature of the core methods used in the tools.

Chapter 7 Development of VOLTTRON™ Compatible AFDD Tool

7.1 Introduction

In this study, the developed AFDD tools are VOLTTRON™ Compatible. The overall structure of the AFDD tools consists of four parts as illustrated in Figure 7-1. The developed VOLTTRON™ AFDD agent is connected to the BAS server of the demonstration building to collect system operation data and to store the collected data into a database. The AFDD algorithms, i.e., WPM-FPCA and WPM-BN methods, are developed in the MATLAB® environment and use the collected BAS operation data stored in the database. An AFDD graphic user interface (GUI) developed in the Microsoft Visual C#® (VC) environment is connected to the MATLAB to present the detection and diagnosis result. Besides, the VC GUI, a web browser-based AFDD demonstration platform is developed to facilitate users to access the AFDD results.

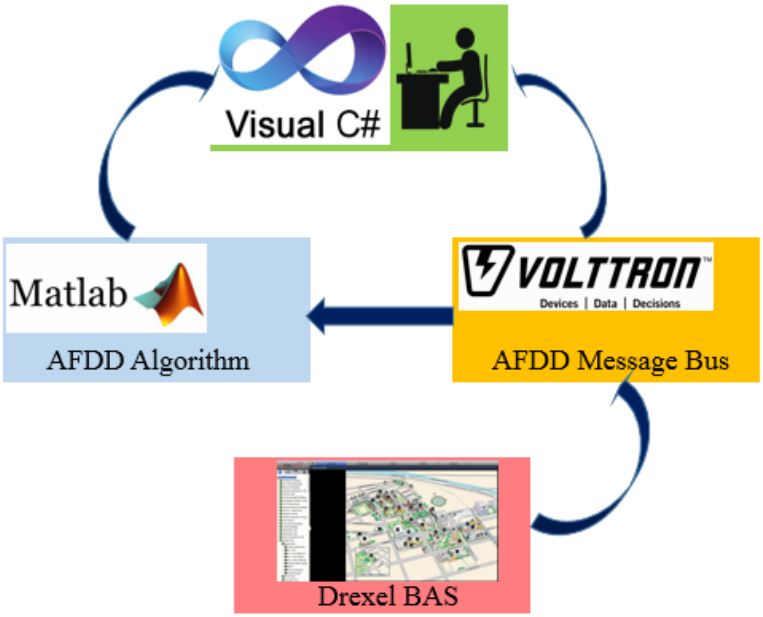


Figure 7-1 Software structure

7.2 Development of VOLTTRON™ AFDD Agent

The VOLTTRON™ AFDD agent is developed using Python 2.7 [112]. The agent includes four main modules, i.e., “Read BAS Measurement Headers”, “Data and Time Control”, “BAS Communication Driver” and “Data Storage Control” as shown in Figure 7-2. The agent can be easily operated on the VOLTTRON™ platform. Through the agent, the BAS data can be collected and published on the VOLTTRON™ platform. A code list is provided in the Appendix III.

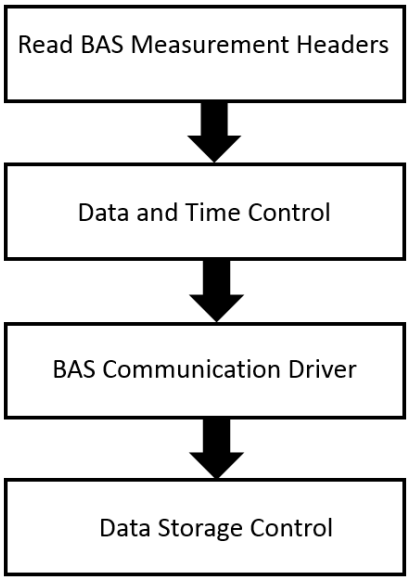


Figure 7-2 AFDD agent working flow

7.3 Development of AFDD GUI

7.3.1 Introduction of Microsoft Visual C#®

Microsoft Visual C#® is a powerful but also easy to use language which aims primarily at developers who create office applications using the Microsoft .NET Framework. VC#® inherits many of the best features of C++ and Microsoft Visual Basic (VB®), there exist few of the inconsistencies and anachronisms which make it more efficient to use C# to develop the interface.

We have used VC# 4.0® and .NET 4.3 version to develop this User Interface. Compared with previous versions of VC# 4.0® and all versions of Visual Basic, this version provides three times more libraries and also much more powerful control components in its .NET Framework with users. It can easily parse data from other program built in other language and Web Clients just by simply adding a specific library in project reference management function, without excessive coding classes or methods for that function. Through using the nearest .NET Framework environment, developer could benefit from all powerful packages and libraries created by professional people for free. These packages are able to offer the most compatible user interface, data access, database connectivity, cryptography, web application development, numeric algorithms, and network communications functions to VC#® users, which result in a much clearer and sampler coding in development process comparing to other developing tools.

Besides, a web browser based demonstration is also developed so that the FDD results can be demonstrated remotely.

7.3.2 VC# based GUI Design

The AFDD GUI developed by the C# is composed of four interfaces including “Login interface” “Fault detection result interface”, and “Fault diagnosis result interface” as shown from Figure 7-3 to Figure 7-5.

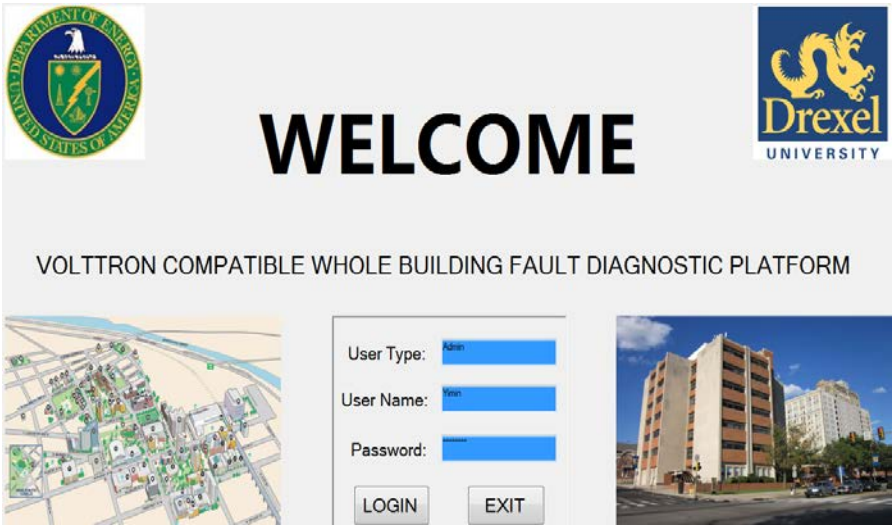


Figure 7-3 Login interface

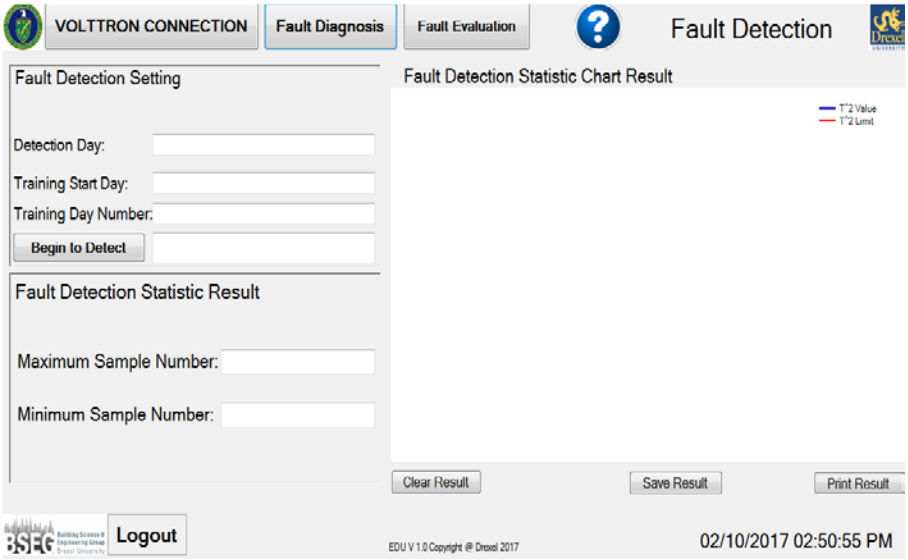


Figure 7-4 Fault detection result interface

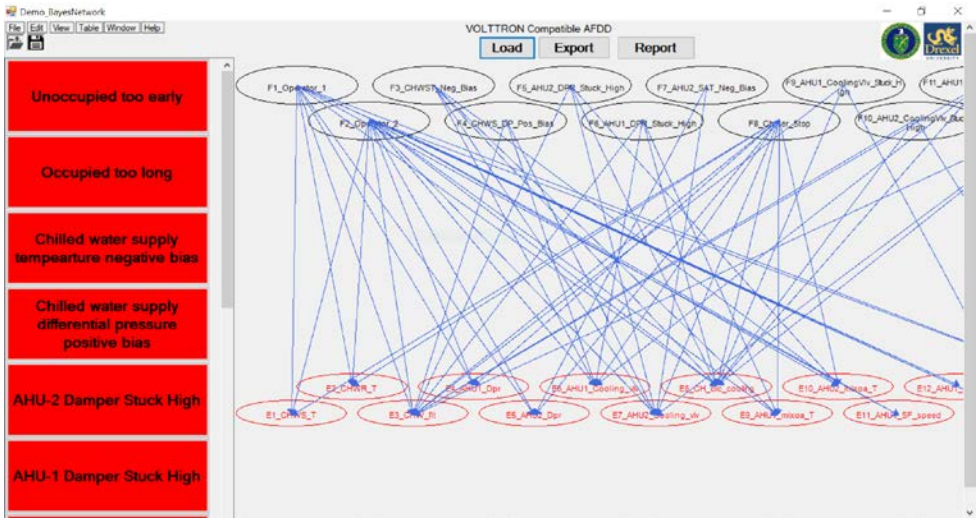


Figure 7-5 Fault diganosis result interface

A VOLTTRON^{TM} and BAS driver interface is developed to facilitate users to set the communication parameters (e.g. downloading date and time) so that the AFDD agent can be connected to the BAS. Data storage path is also set through this interface as shown in Figure 7-6.

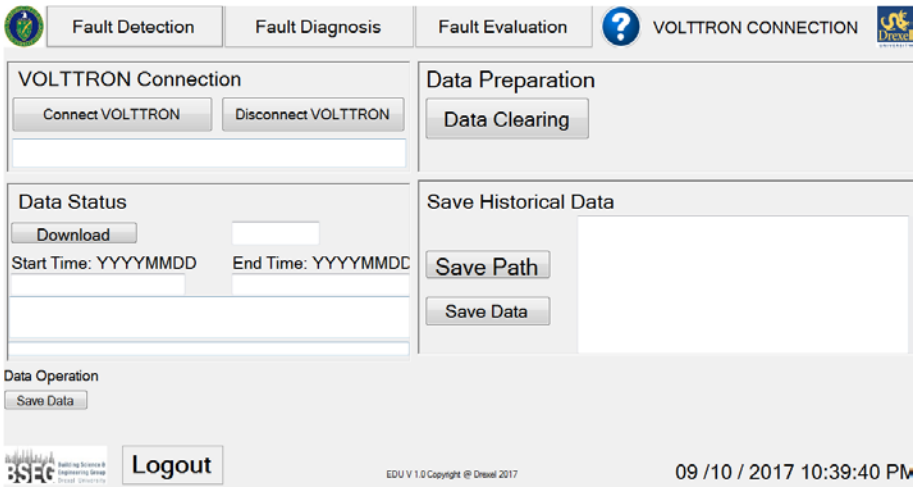


Figure 7-6 VOLTTRON^{TM} connection interface

7.3.3 Web Browser-based GUI Design

A web browser-based AFDD demonstration platform is also developed to help users access the fault detection and diagnosis results remotely. All FDD test cases are stored in the server at Drexel University and can be accessed from a web browser using the following address:

www. bsegafdd.cae.drexel.edu

Figure 7-7 to Figure 7-10 demonstrate each interface designed for the web browser.

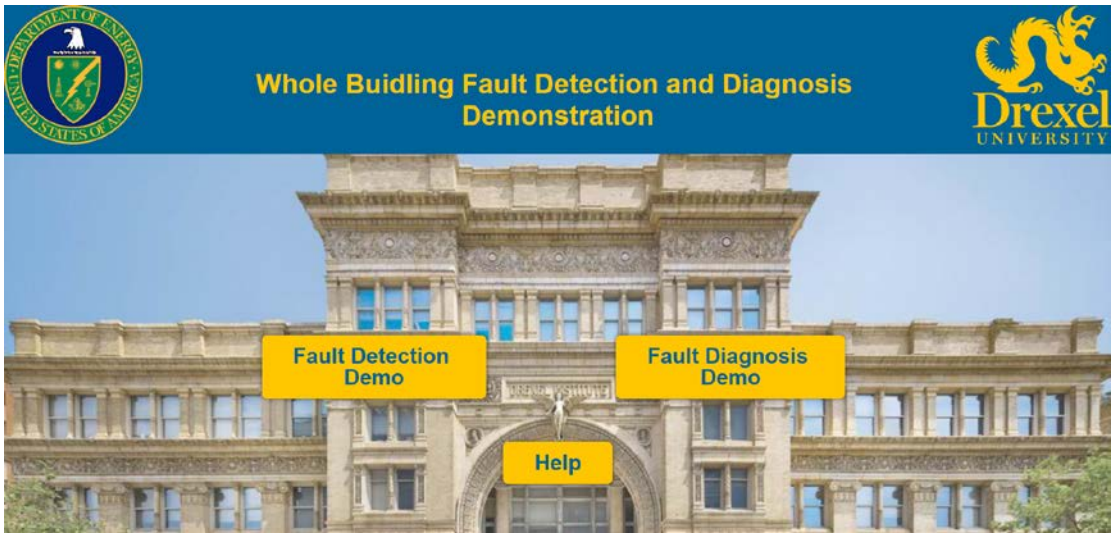


Figure 7-7 Login screen of web-based AFDD demonstration platform

Figure 7-8 shows the developed web-based WPM-FPCA fault detection interface. The interface has two main functions and are divided into two areas for each function. In the first function (left area of the screen), fault detection information is set. In the second function (right area of the screen), fault detection result can be presented using the T^2 statistic graph.

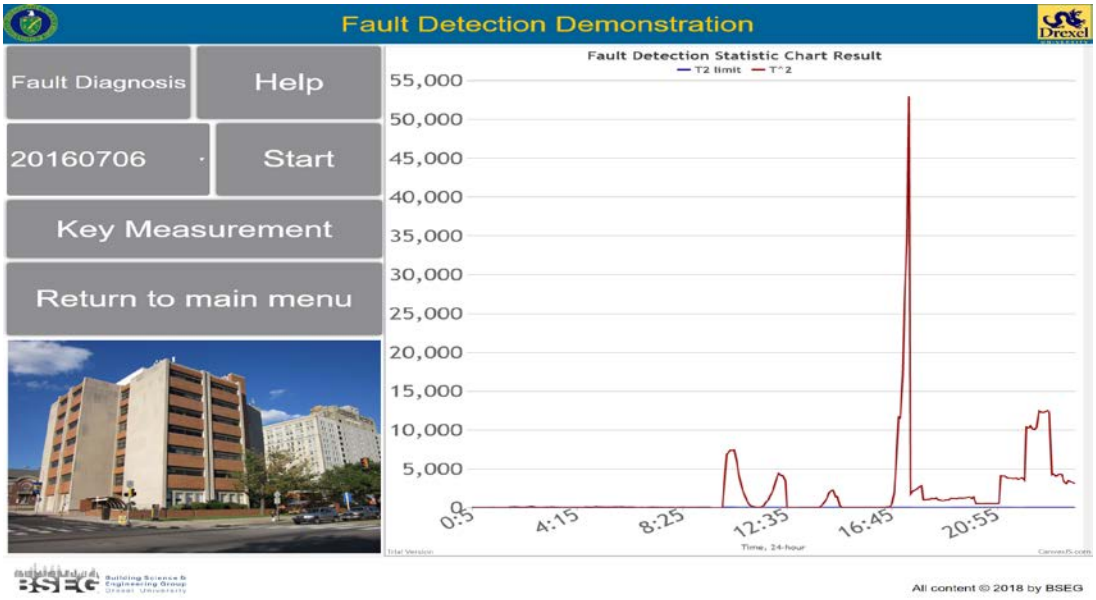


Figure 7-8 Web-based fault detection interface

Figure 7-9 shows the developed web-based WPM-BN diagnosis interface. Similar to the WPM-FPCA detection interface, the left screen area is used to set fault diagnosis information. The right screen area provides the ranked posterior probabilities for each root fault cause.

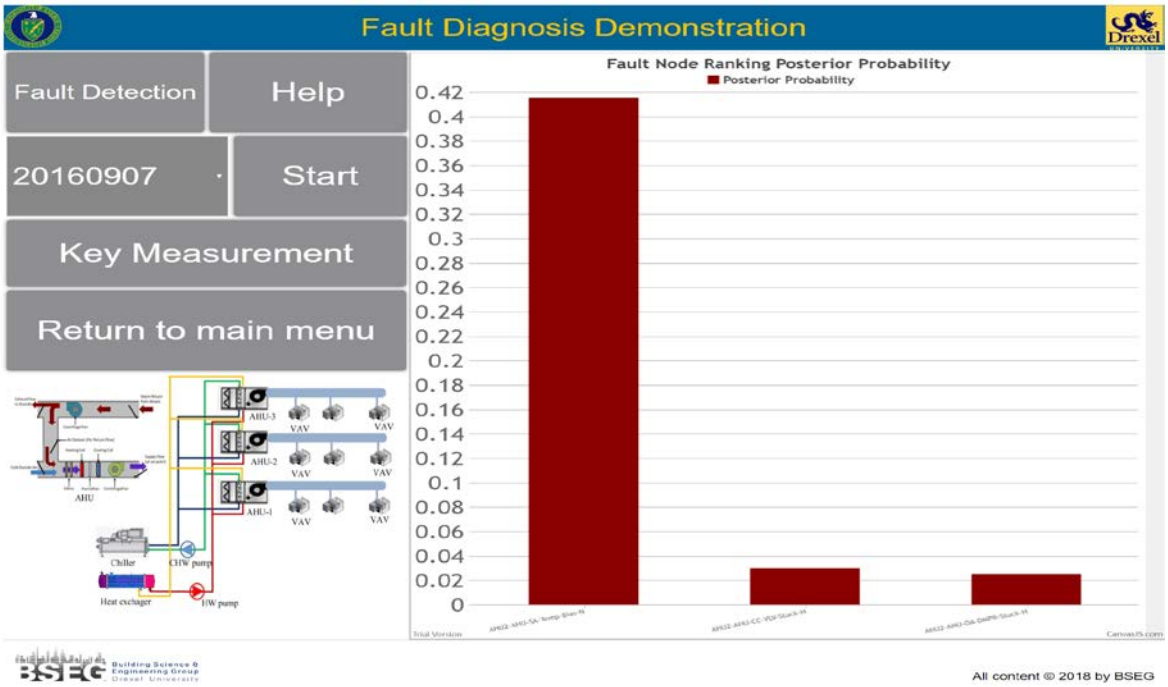


Figure 7-9 Web-based fault diagnosis interface

Figure 7-10 shows the developed web-based key measurement interface. On this interface, selected key measurements from the chiller plant and AHUs are plotted. The time series values for each key measurement from both current and WPM baseline dataset (i.e., under similar weather and building conditions) are compared to help the users to understand the AFDD results and to decide on further maintenance actions.

7.4 Summary

In this Chapter, various components of the developed whole building AFDD tools that are associated with VOLTTRON™ compatibility and interfaces are introduced. These include the development of VOLTTRON™ AFDD agent using Python, the AFDD software GUI using VC#®, as well as the web browser based demonstration platform. Here, the VOLTTRON™ platform is served as a middle ware to connect the core AFDD algorithms with the BAS server. The developed GUIs will facilitate building system operator and other users to identify the system operation performance and locate the fault root-cause.

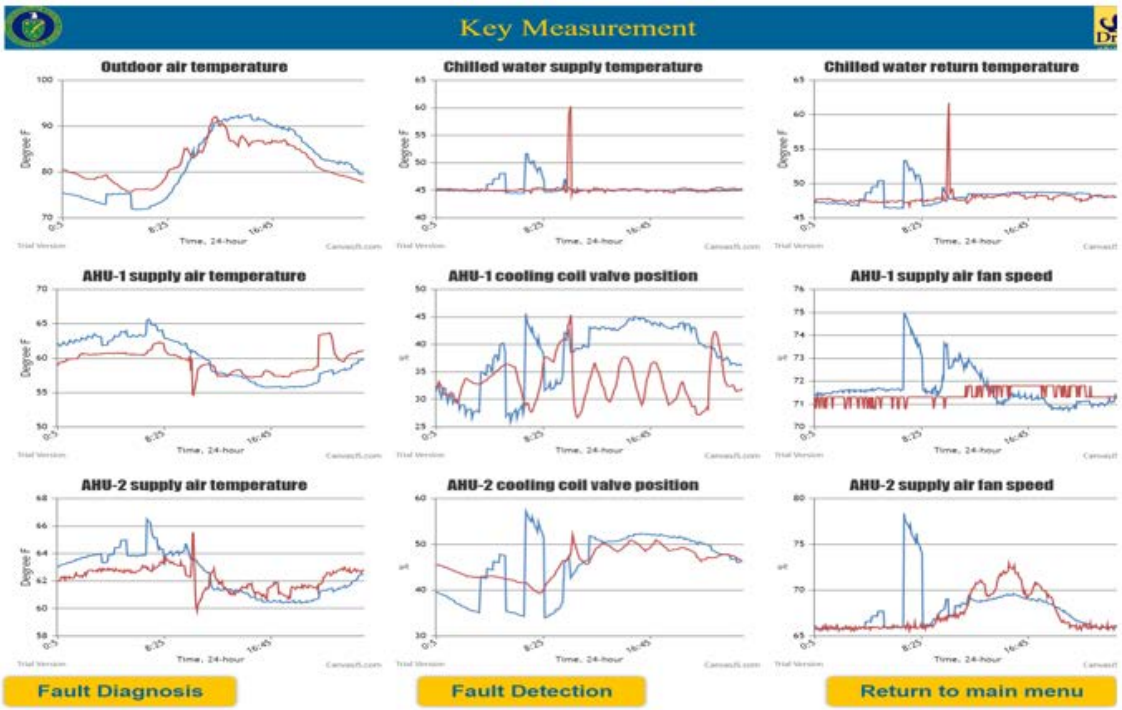


Figure 7-10 Key measurements

Chapter 8 Makeup Fault Test

8.1 Introduction

During the summer of 2018, we have the opportunity to implement more whole building fault tests to further evaluate the performance of the developed WPM-FPCA fault detection method and WPM-BN fault diagnosis method.

Seven faults, which included five supply air subsystem faults, one schedule fault, and one cooling subsystem fault, were implemented during the summer season of year 2018. These faults were selected because of a lack of data collected in previous years that contain these faults.

The collected field test data are analyzed and used to evaluate the developed AFDD methods. Details about the analysis and evaluation are provided in Section 8.2. Besides the implemented faults, other naturally occurred faults were observed in multiple subsystems for most of the test days.

8.2 Fault Tests and Method Evaluation

8.2.1 Fault Test Summary

Table 8-1 summarizes the details of the implemented faults. System operation and abnormalities caused by either the implemented fault, or naturally occurred faults are also summarized. Impacts from the artificially implemented and naturally occurred faults are observed using the same manual method as described in Chapter 6 by comparing test day data with baseline data selected using the WPM method.

As observed from Table 8-1, among the seven test cases, the ground truth of one test case (AHU-2 SA static pressure sensor positive bias 0.2 inch of water on July 19th) was not established because of a lack of baseline data with similar weather conditions. WPM-FPCA method failed to detect any abnormality therefore. For the other six test cases, whole building abnormalities, either caused by the implemented fault or by other naturally occurred faults in different subsystems, were observed. Using the developed WPM-FPCA and WPM-BN AFDD tools, the FDD results are summarized in the last three columns in Table 8-1. For the WPM-FPCA method evaluation, results from both 0.65 and 0.95 retention rates were compared.

8.2.2 Fault Test Description and FDD Result

Detailed fault test description, data analysis, and FDD results are given in the following sections.

Table 8-1 Fault test and FDD results

Date	Time		Whole Building Fault Description				FDD Result		
	Start	End	Fault Description	Implementation	Impacts from Implemented Fault	Other Abnormalities Caused by Simultaneous Naturally occurred Faults	WPM-FPCA (Retention Rate - 0.65)	WPM-FPCA (Retention Rate - 0.95)	WPM-BN Diagnosis Result (Top three faults with high posterior probabilities)
07/09/18	10:15	20:15	AHU-2 supply air temperature sensor bias fault negative 3.5°F	Override supply air temperature setpoint demand adj by 1.75	AHU-2 SA temperature is higher than the baseline; AHU-2 mixed air temperature is higher than the baseline	Chiller CHW differential pressure is higher than the baseline; AHU-2 SA fan speed higher than the baseline; AHU-1 cooling coil valve position is higher than the baseline; AHU-3 supply air fan speed is higher than the baseline; AHU-3 supply air pressure is lower than the baseline	detected	detected	1) AHU-2 OA damper stuck at a higher position fault: 79.8% 2) AHU-1 OA damper stuck at a higher position fault: 60% 3) AHU-1 cooling coil valve operate at higher position fault: 32.9% Implemented fault: AHU-2 SA temperature negative bias fault: 18.2%
07/10/18	10:30	20:30	AHU-2 OA damper stuck at 30% open (higher than the normal position -15% open)	Override OA damper at 30% open in the BAS	AHU-2 cooling coil valve and mixed air temperature are higher than the baseline, Chiller CHW flowrate is higher than the baseline	Chiller CHW differential pressure is higher than the baseline; AHU-1 cooling coil valve position higher than baseline, AHU-1 supply air temperature lower than the baseline	NOT detected	detected	1) AHU-2 OA damper stuck at a higher position fault: 40.8% 2) AHU-1 OA damper stuck at a higher position fault: 12% 3) AHU-1 cooling coil valve operate at higher position fault: 11.2%
07/11/18	10:00	20:00	AHU-2 cooling coil valve stuck at 80%(higher than the normal position -40-60% open)	Override cooling coil valve control signal at 80% open in the BAS	AHU-2 SA temperature lower than baseline; cooling coil valve control signal lower the baseline; delivered cooling is higher than the baseline	AHU-1 cooling coil valve position is higher than baseline, AHU-1 supply air temperature is lower than the baseline	detected	detected	1) AHU-2 cooling coil valve stuck at a higher position fault: 87.3 % 2) AHU-1 cooling coil valve operate at higher position fault: 42.1% 3) AHU-1 cooling coil valve stuck at higher position fault : 12.6 %
07/18/18	9:30	19:30	AHU-2 OA damper stuck at 60% open (higher than the normal position -15% open)	Override OA damper at 60% open in the BAS	AHU-2 SA cooling coil valve higher than baseline; AHU-2 mixed air temperature is higher than baseline; significant delivered cooling loss	AHU-1 SA static pressure sensor is lower than the baseline	detected	detected	1) AHU-2 OA damper stuck at a higher position fault: 62.4% 2) AHU-1 cooling coil valve operate at higher position fault: 22.6% 3) AHU-2 cooling coil valve operate at higher position fault : 17.1%
07/19/18	10:15	20:30	AHU-2 SA static pressure sensor positive bias 0.2 H2O inch	Override SA static pressure from 1.8 to 2.0 inH2O	Failed test because no enough baseline sample generated				

Table 8-1 Fault test and FDD results (Cont'd)

Date	Time		Whole Building Fault Description				FDD result		
	Start	End	Fault Description	Implementation	Impacts from Implemented Fault	Other Abnormalities Caused by Simultaneous Naturally occurred Faults	WPM-FPCA (Retention Rate - 0.65)	WPM-FPCA (Retention Rate - 0.95)	WPM-BN Diagnosis Result (Top three faults with high posterior probabilities)
07/22/18	20:20	21:40	Change weekend occupied schedule to end at 8:20 PM (earlier unoccupied)	Stop the HVAC system at 8:20PM	Multiple impacts on AHU-1 to 3	Multiple impacts on AHU-1 to 3	detected	detected	1) System operation schedule unoccupied fault: 99.9% 2) AHU-1 SA temperature sensor negative bias fault: 67.2% 3) Chiller stop fault: 45.3%
07/23/18	8:00	18:00	CHWS temperature negative bias 3.0 °F	Change CHW supply temperature setpoint from 44 °F to 47°F	AHU-1 cooling coil valve is lower than the baseline; AHU-1 supply air static pressure is lower than the baseline	AHU-2 speed is lower than the baseline; AHU-2 SA pressure is higher than the baseline; AHU2 cooling coil valve does not react	NOT detected	detected	1) AHU-1 SA temperature sensor negative bias fault: 33.1% 2) System operation schedule unoccupied fault: 30.0% 3) Chiller stop fault: 11.5%

8.2.2.1 Fault Test on July 9th 2018

On July 9th, a supply air temperature sensor bias fault was implemented. AHU-2 supply air temperature sensor with a negative bias (3.5 °F) was implemented by overriding the supply air demand adjust point from 0 to 1.75. This negative bias fault was expected to cause the actual AHU supply air temperature to be higher than the screen value. Therefore, the damper positions of the downstream VAV terminal units would be higher than their baseline values, in order to allow more air flow to enter the zone and make up the increased supply air temperature. The supply air static pressure would be disturbed, and the AHU supply air fan would run at a higher speed to increase the static pressure. From the operation data during the test and from baseline as shown in Figure 8-1, it is observed that the AHU-2 supply air static pressure was lower than the baseline, and the supply air fan speed was higher than the baseline.

Besides the implemented fault, abnormalities caused by multiple faults which naturally occurred were also observed. The chilled water differential pressure was higher than the baseline, as shown in Figure 8-2. In the AHU-1, cooling coil valve position was observed to be higher than the baseline, and the supply air static pressure was observed to be lower than baseline, as show in Figure 8-3.

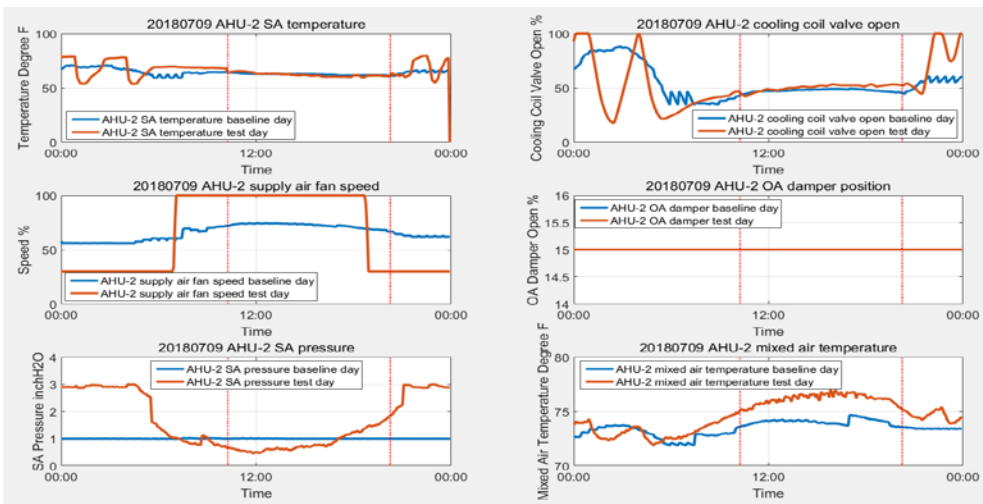


Figure 8-1 AHU-2 operation between the test day and the baseline (July 9th, 2018)

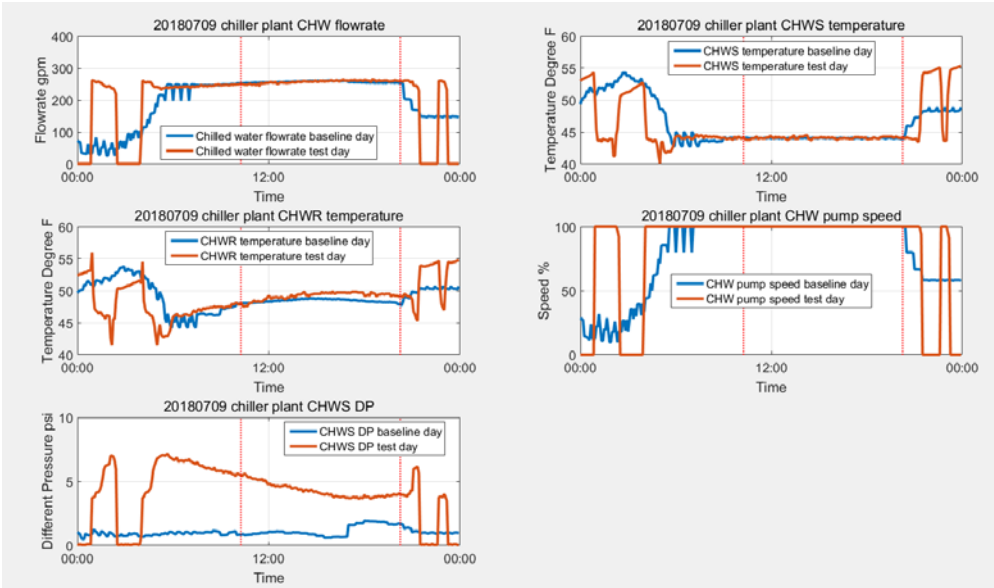


Figure 8-2 Chiller plant operation between the test day and the baseline (July 9th, 2018)

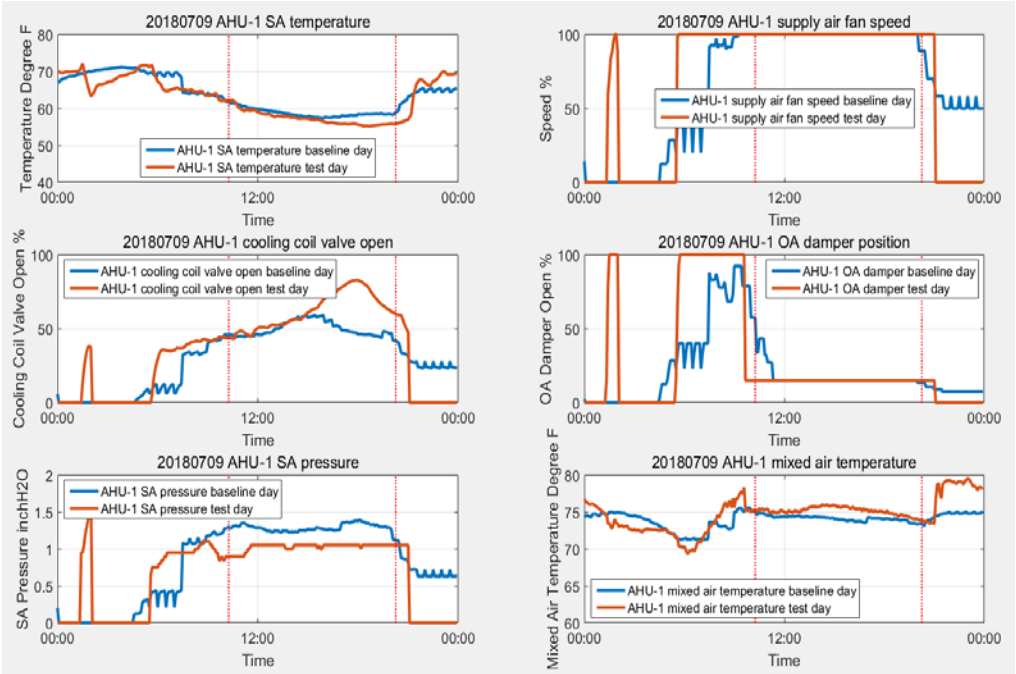


Figure 8-3 AHU-1 operation between the test day and the baseline (July 9th, 2018)

The whole building abnormality was successfully detected using both 0.65 and 0.95 retention rates as shown in Figure 8-4 (with a retention rate of 0.95).

Due to multiple faults occurred in different systems, this is a challenging fault isolation case. Because a lack of ground truth (we are unsure of the exact faults that have caused the abnormalities in other subsystems), it is hard to quantitatively evaluate the BN fault diagnosis result. As shown in

Figure 8-5, WPM-BN method's result shows that the three faults with higher posterior probability are: 1) AHU-2 OA damper stuck at a higher position fault: 79.8%; 2) AHU-1 OA damper stuck at a higher position fault: 60%; and 3) AHU-1 cooling coil valve operate at higher position fault: 32.9%. While, the Implemented fault, AHU-2 SA temperature negative bias fault, has the posterior probability of 18.2%. These diagnosed faults would lead to the abnormalities as we have observed. Without additional system examination, it is difficult to isolate the exact faults that actually occurred.

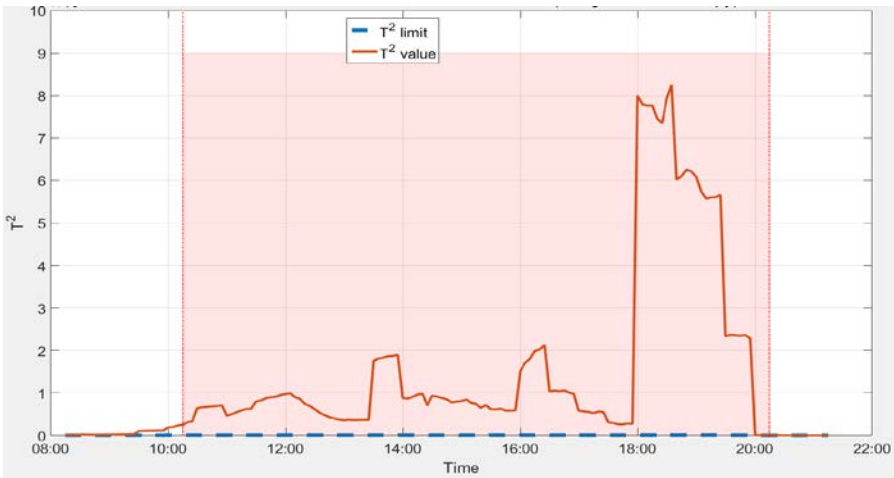


Figure 8-4 Fault detection result under 0.95 PC retention rate (July 9th, 2018)

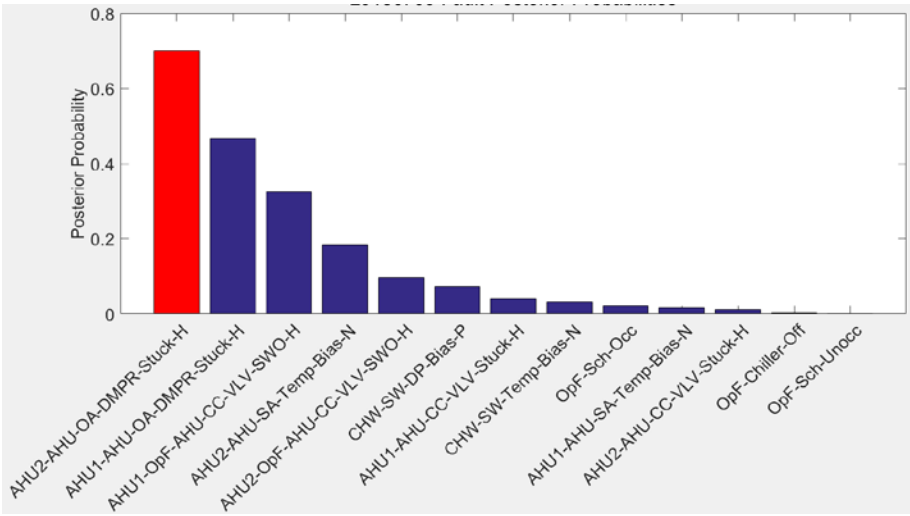


Figure 8-5 Fault diagnosis result (July 9th, 2018)

8.2.2.2 Fault Tests on July 10th and July 18th 2018

On July 10th and July 18th 2018, the outdoor air damper at AHU-2 was artificially stuck at 30% and 60% opening by overriding the corresponding control signal in the BAS. The stuck positions (30% and 60% open) were higher than the damper's normal position (15% open) under similar weather conditions. In the summer, the HVAC system of the demo building is operated under cooling mode, where the AHU outdoor air damper is typically controlled to be at a 15% opening, in order to maintain the minimum requirement of fresh air flowrate. However, during the early morning and late night, where the outdoor enthalpy is lower than the return air enthalpy, the outdoor air damper is controlled under economizer mode to save energy. On these two days, the fault test period was from 10:30 to 20:30 and 09:00 to 19:30, respectively.

On July 10th, besides the implemented fault, other abnormalities caused by other naturally occurred faults in different systems were also observed. For example, in the AHU-1, the cooling coil valve position was higher than the baseline, and supply air temperature was lower than the baseline, as shown in Figure 8-6.

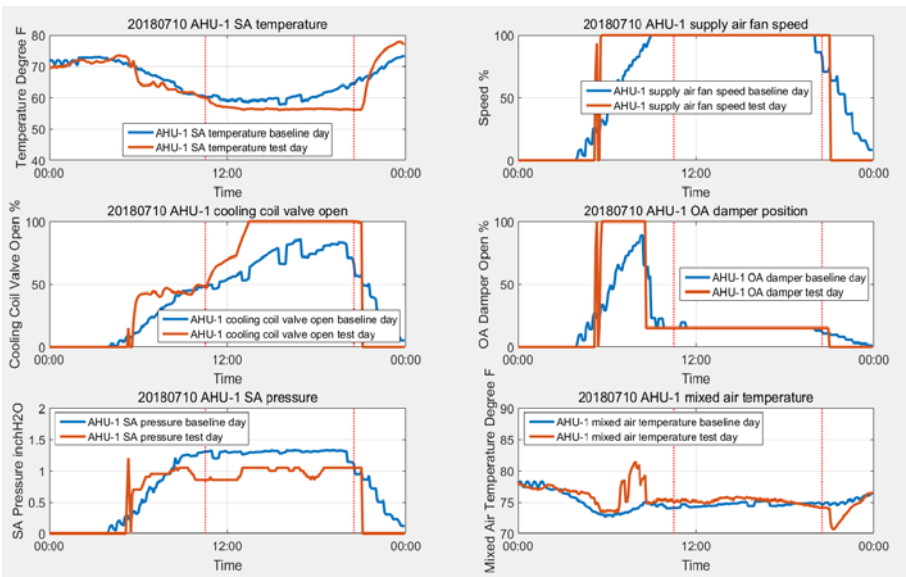


Figure 8-6 AHU-1 operation between the test day and the baseline (July 10th, 2018)

On July 18th, besides the implemented fault, AHU-1 SA static pressure was lower than the baseline. And AHU-1 cooling coil valve position was higher than the baseline in some time period during the fault test, as shown in Figure 8-7.

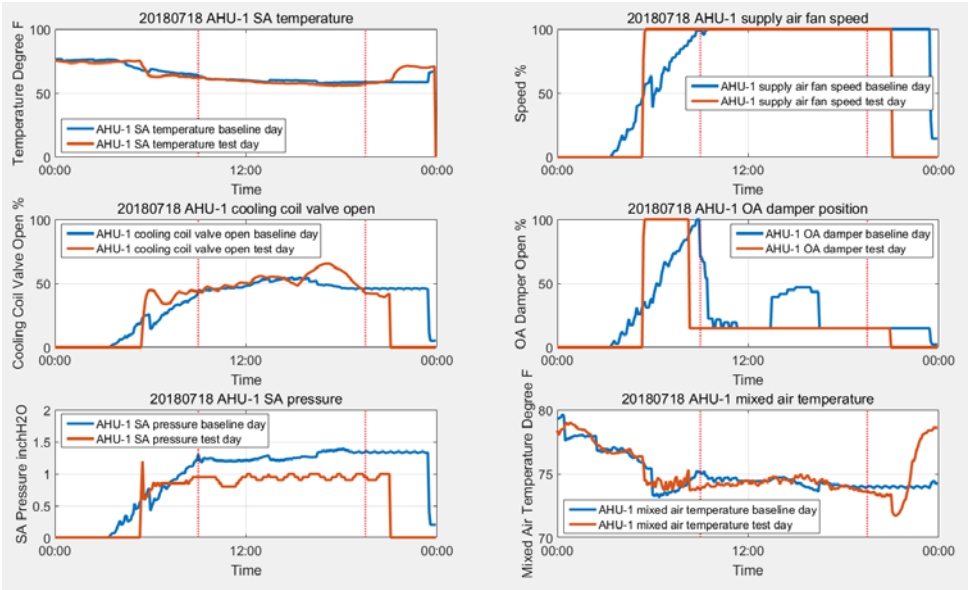


Figure 8-7 AHU-1 operation between the test day and the baseline (July 10th, 2018)

Both of these two test days were detected as abnormal when using the WPM-FPCA fault detection method with a retention rate of 0.95 as shown in Figure 8-8 and Figure 8-9. When using a retention rate of 0.65, the test case with a milder fault severity (30% open) was not detected.

WPM-BN diagnosis results for both cases are shown in Figure 8-10 and Figure 8-11. According to the posterior probabilities, the WPM-BN method ranked the implemented fault as the top potential fault.

On July 10th, it shows that the three faults with higher posterior probability are: 1) AHU-2 OA damper stuck at a higher position fault: 40.8%; 2) AHU-1 OA damper stuck at a higher position fault: 12%; and 3) AHU-1 cooling coil valve operate at higher position fault: 11.2%.

On July 18th, it shows that the three faults with higher posterior probability are: 1) AHU-2 OA damper stuck at a higher position fault: 62.4%; 2) AHU-1 cooling coil valve operate at higher position fault: 22.6%; and 3) AHU-2 cooling coil valve operate at higher position fault: 17.1%.

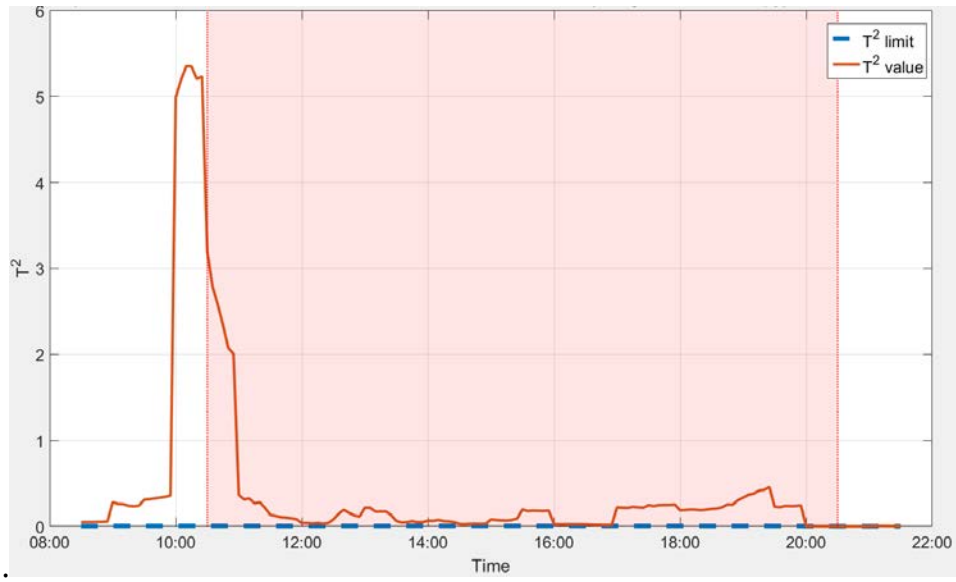


Figure 8-8 Fault detection result under 0.95 PC retention rate (July 10th, 2018)

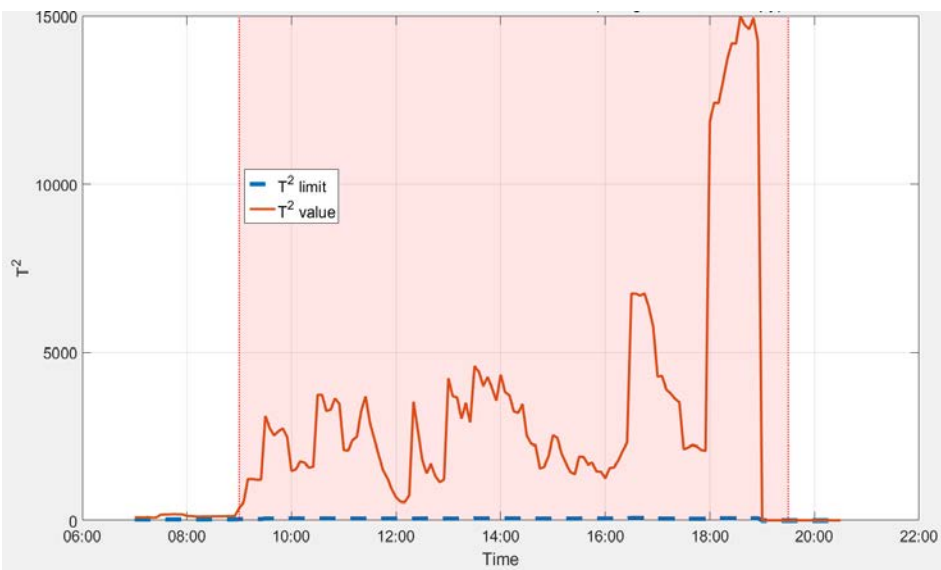


Figure 8-9 Fault detection result under 0.95 PC retention rate (July 18th, 2018)

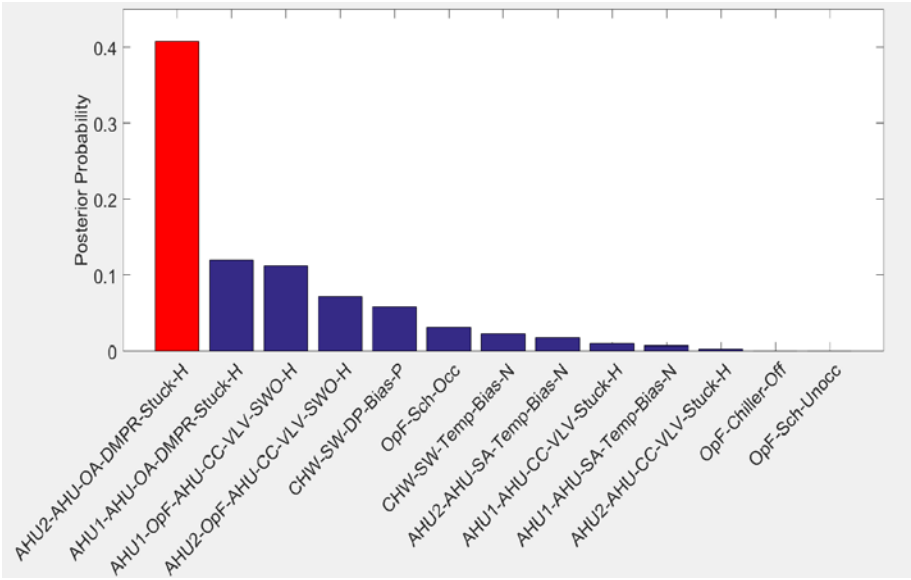


Figure 8-10 Fault diagnosis result (July 10th, 2018)

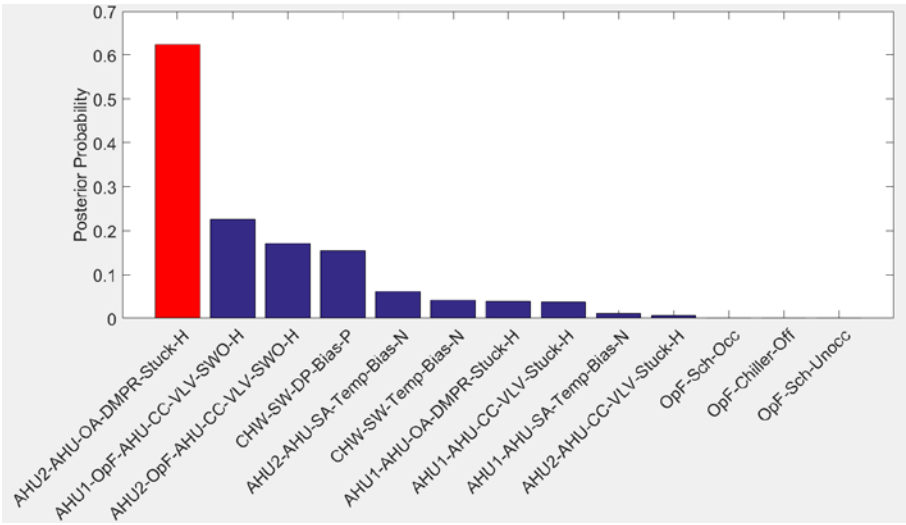


Figure 8-11 Fault diagnosis result (July 18th, 2018)

8.2.2.3 Fault Test on July 11th, 2018

In this day, an AHU-2 cooling coil valve fault (stuck at a higher than normal position) was implemented. The AHU-2 cooling coil valve was stuck at 80% open position, which was higher than the normal position (40% to 60%). The fault test period was from 10:00 to 20:00. As expected, whole building fault symptoms were observed. The AHU-2 supply air temperature was lower than the normal value, and thus caused the downstream VAV terminal units to reduce their discharge

air flow rates, which further led to the supply fan speed to be lower than the baseline value as shown in Figure 8-12.

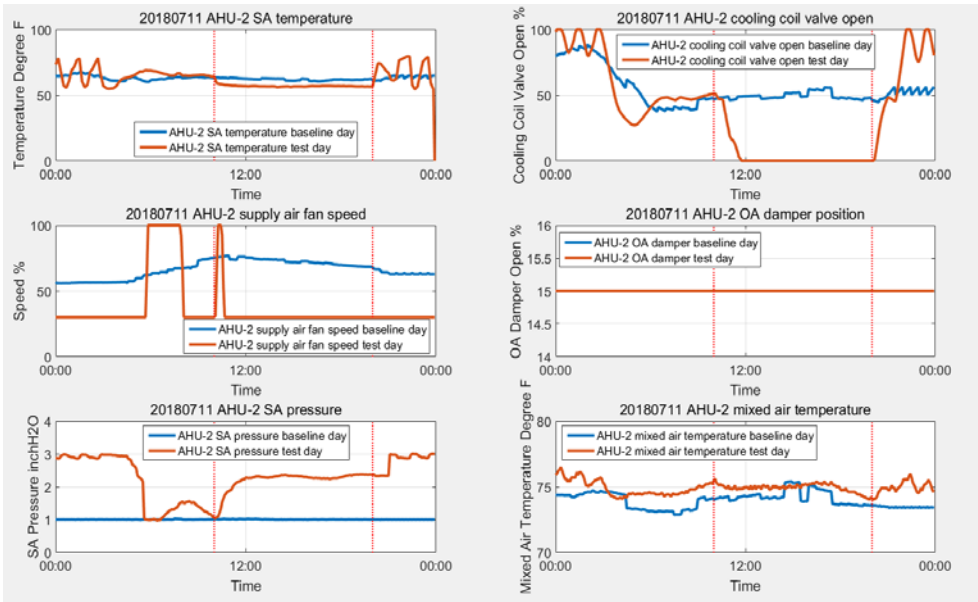


Figure 8-12 AHU-2 operation between the test day and the baseline (July 11th, 2018)

Besides the implemented fault, AHU-1 cooling coil valve position was observed to be higher than baseline. The supply air temperature and the static pressure in AHU-1 were observed to be lower than the baseline as shown in Figure 8-13.

The whole building abnormality was successfully detected as shown in Figure 8-14 (with retention rate of 0.95).

The diagnosis result (Figure 8-15) shows that three faults were diagnosed with high posterior probabilities: 1) AHU-2 cooling coil valve stuck at a higher position fault: 87.3 %; 2) AHU-1 cooling coil valve soft-ware-override at higher position fault: 42.1%; and 3) AHU-1 cooling coil valve stuck at higher position fault: 12.6 %. Again, these diagnosed faults agree well with the observed fault symptoms.

8.2.2.4 Fault Test on July 22nd, 2018

A schedule fault, i.e., system is unoccupied at an earlier than normal time, was implemented by stopping the HVAC system at 20:20 (normal time is 21:00). This fault impacted multi-subsystems. This fault, which demonstrated whole building fault symptoms, was successfully detected and diagnosed by the developed FDD method, as show in Figure 8-16 and Figure 8-17.

The diagnosis result shows that the three faults with higher posterior probability are: 1) system operation schedule unoccupied fault: 99.9%; 2) AHU-1 SA temperature sensor negative bias fault: 67.2%; and 3) chiller stop fault: 45.3%.

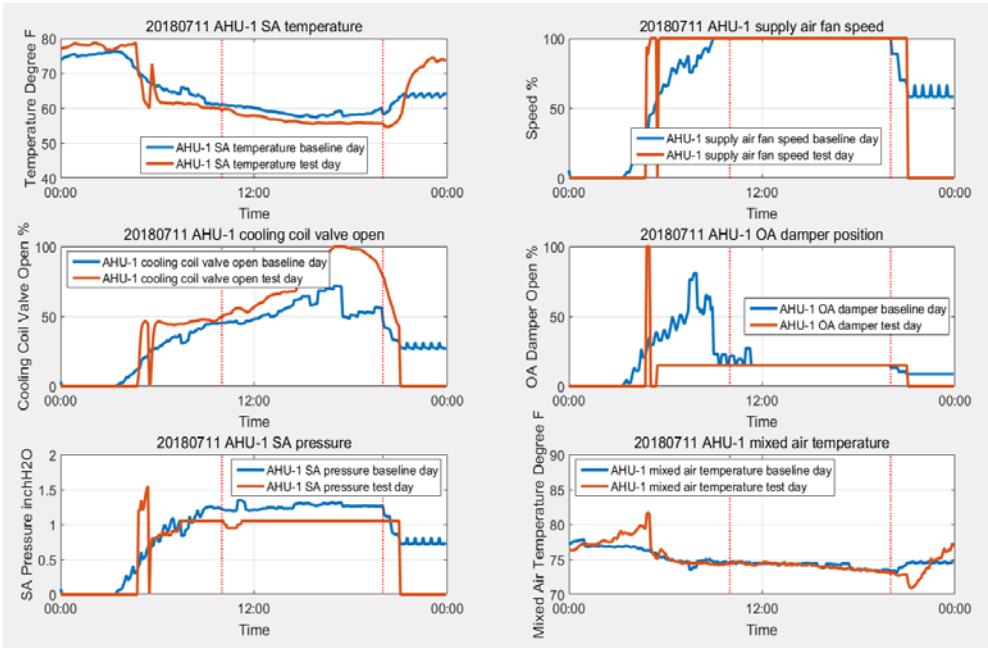


Figure 8-13 AHU-1 operation between the test day and the baseline (July 11th, 2018)

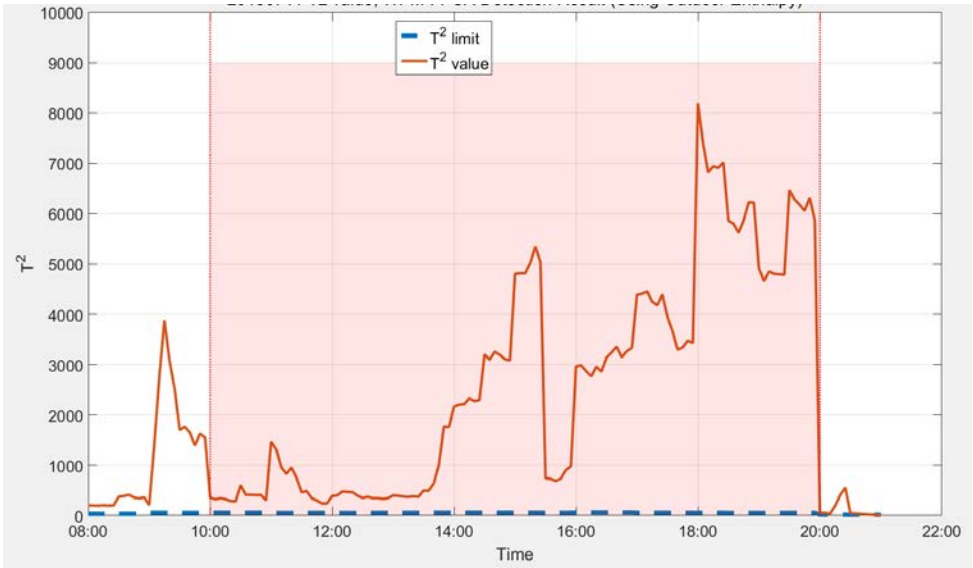


Figure 8-14 Fault detection result under 0.95 PC retention rate (July 11th, 2018)

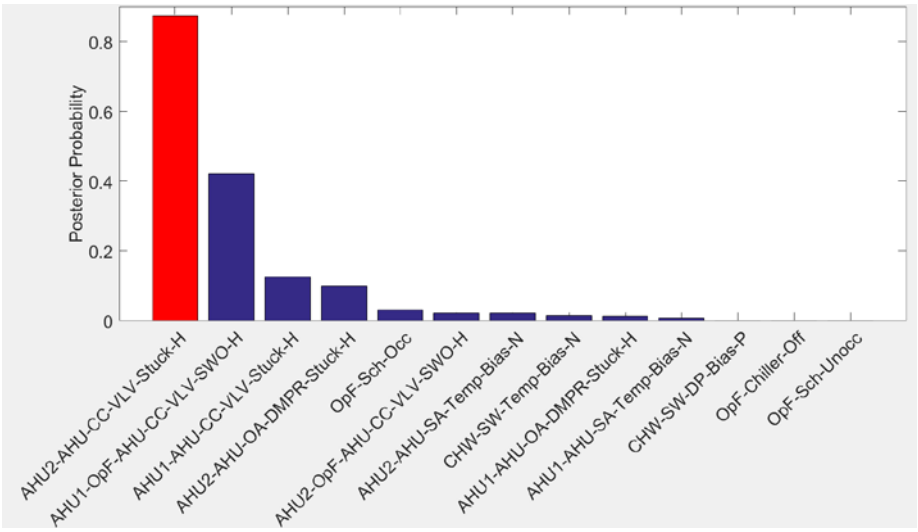


Figure 8-15 Fault diagnosis result (July 11th, 2018)

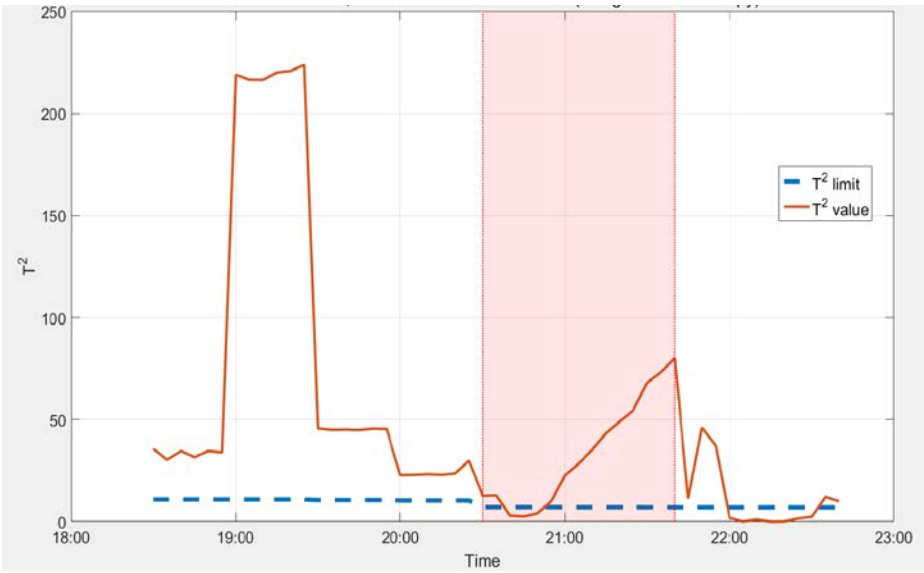


Figure 8-16 Fault detection result under 0.95 PC retention rate (July 22nd, 2018)

8.2.2.5 Fault Test on July 23rd, 2018

A sensor bias fault was implemented on the chilled water supply temperature sensor. On this day, the bias was set to be negative 3°F through adjusting the outlet chilled water temperature setpoint from 44°F to 47 °F.

This fault caused the AHU-2 cooling coil valve position to be lower than baseline, as shown in Figure 8-18.

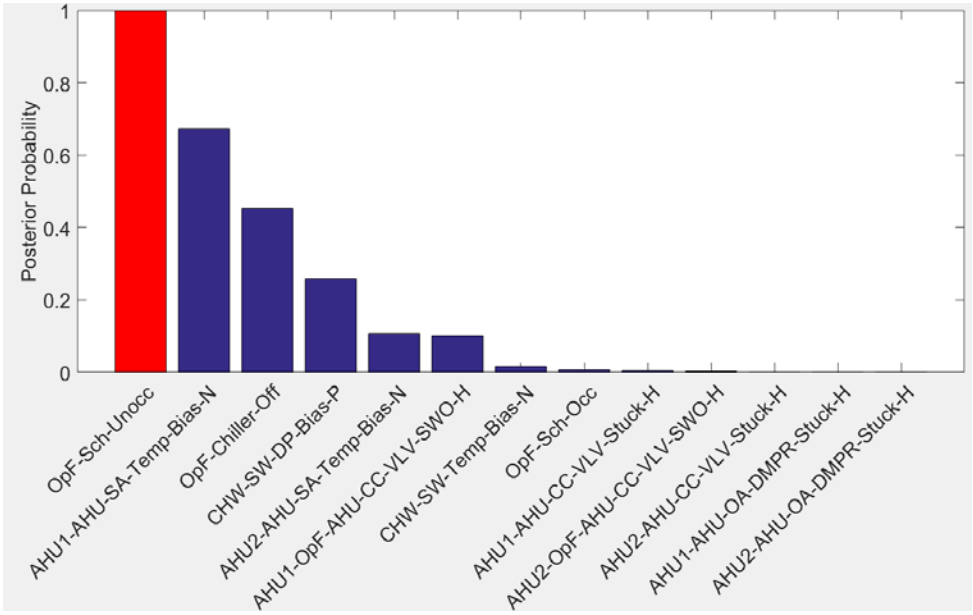


Figure 8-17 Fault diagnosis result (July 22nd, 2018)

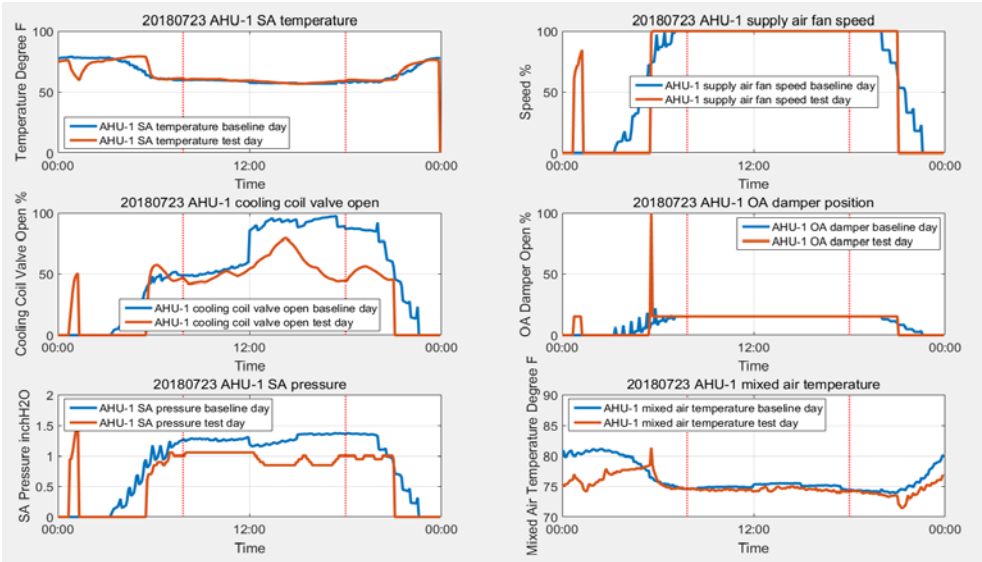


Figure 8-18 AHU-1 operation between the test day and the baseline (July 23rd, 2018)

At the same time, some other abnormalities were observed in the AHU-2. For example, supply air fan speed was lower than the baseline, and supply air pressure was higher than the baseline as shown in Figure 8-19. This may be caused by the faults from the downstream VAV terminal units, or different zone loads.

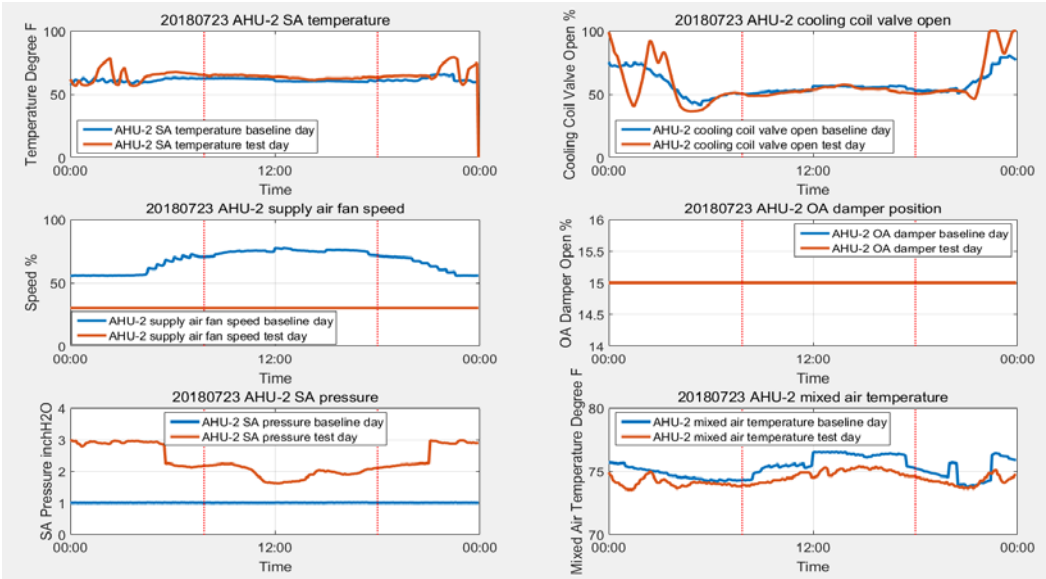


Figure 8-19 AHU-2 operation between the test day and the baseline (July 23rd, 2018)

As there were abnormalities in different systems, this day was successfully detected by the WPM-FPCA method as shown in Figure 8-20 (with a retention rate of 0.95). But since different fault root-causes may lead to the fault symptoms, the posterior probability of the implemented fault was not calculated as the top one form the WPM-BN method as shown in Figure 8-21. The diagnosis result shows that the three faults with higher posterior probability are: 1) AHU-1 SA temperature sensor negative bias fault: 33.1%; 2) System operation schedule unoccupied fault: 30.0%; and 3) Chiller stop fault: 11.5%. The implemented fault was diagnosed a potential fault with a posterior probability of around 8%.

8.3 Summary

In this Chapter, the seven makeup fault tests implemented during summer of 2018 are summarized. The developed AFDD tools are further evaluated using the makeup test data. These faults have been selected due to a lack of collected data that contain these faults from previous tests. Simultaneous naturally occurred faults have been observed in nearly all cases. The developed WPM-FPCA method has been demonstrated to be effective at detecting whole building abnormalities caused by the implemented faults and/or multiple simultaneous naturally occurred faults. However, it should be noted that the naturally occurred faults have made it difficult to establish the ground-truth to evaluate the WPM-BN fault diagnosis method. Despite of this difficulty, the WPM-BN fault diagnosis method has successfully demonstrated that it is able to suggest potential faults that could have caused the observed abnormalities.

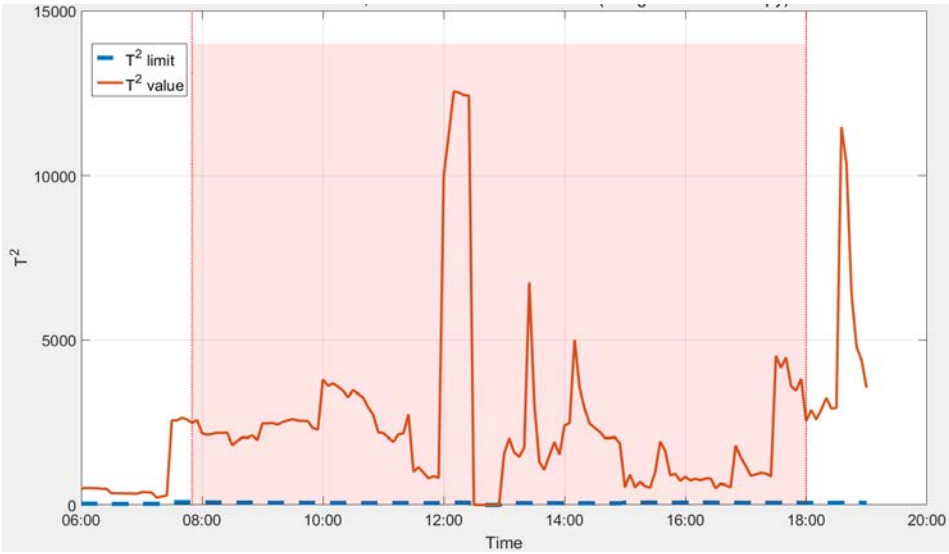


Figure 8-20 Fault detection result under 0.95 PC retention rate (July 23rd, 2018)

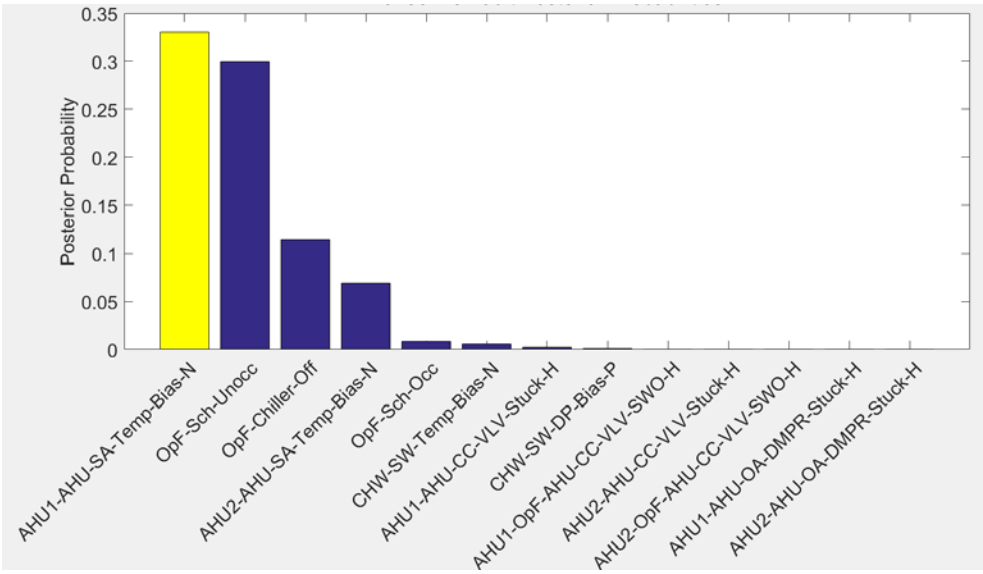


Figure 8-21 Fault diagnosis result (July 23rd, 2018)

Chapter 9 Market Study

9.1 Market and Competition

Years ago, fault diagnostic technologies were tested and trusted by owners who later found the software unreliable. Since then, several companies have worked to address some of the most glaring flaws of the AFDD technology, including creating two systems of fault detection (rules and symptoms), new methods of installation (including modern retro-commissioning), and diagnostic assessment (sub-automatic fixing and automatic fixing). The result is a market of diverse AFDD technologies that focus on niche markets. Current market adoption is limited, but in our survey of service providers, most expressed a shared prediction that prices within the industry will fall over the next two years as effectiveness of the technologies available increases.

Our AFDD software is different than the above methods so that once users establish an expected return on investment and limit false-detections, users of the technology can expect a holistic improvement to their facilities.

It is estimated that 15 to 30% of energy is wasted in commercial buildings due to various faults in HVAC equipment and controls. Since the market for AFDD software is relatively new, consumer interest and education is low. The technology requires a robust marketing presence to educate consumers and drive market adoption that is most easily provided through larger companies. There are currently no benchmarking or performance standards nor government coding standards. This disincentives widespread market adoption of the fault diagnostic software because they are difficult to compare and not legally required in most buildings. Without well-defined parameters for evaluating product performance, it can be difficult for consumers to make a well-informed decision, leading to minimal technology adoption.

The current market is poised for significant growth with increased market adoption in the next decade, which is why this is an excellent time to introduce a cutting-edge technology. According to a report conducted by Navigant Research, the Building Energy Management Systems industry is predicted to quintuple over the course of nine years with most of the growth being concentrated in Europe, Asia, and North America. The report details how over a four year period between 2010 and 2014 the industry was led by the creation of several BEMS-focused startups and how the next nine years will see a transition from startups to acquisitions as individual companies come to dominate the industry and will find the most profit in expanding their portfolios. Predictably, future AFDD companies will seek a purchasing option to best take advantage of the increased profit potential that an acquisition-focused market brings.

9.2 Distribution Recommendations

There are several big players in the AFDD software market that create significant barriers to gaining market share. Creating a business around the software would be costly and time-consuming. Additionally, the Navigant report indicates that these companies will be looking for new technologies to serve the needs of an industry that will likely quintuple in the next decade. For these reasons, we recommend licensing the technology to the major companies that already

provide fault diagnostic services for HVAC systems. Companies that have a well-established client base, exhibits strong growth in the areas of fault detection services, and has previously licensed technology are optimal for a licensing agreement. We believe that larger service providers will best fit these characteristics.

There are three primary types of distributors who would be strategic partners in a licensing deal. The first type is companies that sell and maintain building control systems. Siemens and Allen-Bradley hold significant market share in this segment and license existing technology they use. Companies of this sort generally have capital flows of millions of dollars, giving them the capital to implement retro-commissioned software and enter into a more expensive licensing agreement. While they may have the incentive to purchase AFDD technology, many of these companies are already investing in their own research teams to develop inner-company brands of diagnostic software. Their R&D departments are well funded and produce excellent software. We believe that the robust nature of the R&D departments will actually incentivize them to license the software since they have engineers who have the time and funding to improve upon and best implement the technology.

The second type of company we'll target includes companies that act as national facilitators, such as Emcor and Comfort Systems. They focus on maintaining and operating buildings for clients. Companies of this sort have already developed their own client networks and have demand for software that increases their profits and efficiency. These companies may also be less likely to have a robust R&D team, which necessitates licensing agreements to have the best software available to their clients. One of the benefits of licensing with one of these companies is their national network is large enough that initial rollout of a new AFDD software can be relatively large.

The third type includes companies that perform data analytics and mining that presents information to its owners, such as Building IQ. These companies do not develop or install their own fault detection software, so they would benefit greatly from having a licensing agreement. However, their capital and cash flow is less than the above companies, which means the licensing agreement would have to be at a lower price point. These companies function similarly to national facilitators, as they also have the benefit of their own client networks while trusting their service to an externally developed fault detection and diagnostic software.

9.3 Survey Study

A survey study was performed to understand the existing AFDD market, including its gap and need. The above market and distribution discussions are based on the survey study findings. The details of this survey study are provided in Appendix IV.

https://drexel.qualtrics.com/SE/?SID=SV_6ujZWI0IPOJf4r3

https://drexel.qualtrics.com/SE/?SID=SV_78rH3vujFJ5PlKB

Chapter 10 Undergraduate Engagement

From year 2016 to year 2018, a total of twelve Drexel undergraduate students have been involved in this project to perform different research tasks. Their names and responsibilities are provided in Table 10-1. Among them, Ojas Pradhan will pursue a doctoral degree in building energy efficiency area. Ojas Pradhan and Noor Solatch have been awarded by the Sustainable Energy Fund to attend the 2018 EnergyPath Conference. Two students are currently working in the building industry pursuing energy efficient design and operation strategies. One student has been employed by a Big Data enterprise company, benefited from the experiences he has gained during this project.

Table 10-1 Undergraduate Engagement

Year	Name	Major	Responsibility
2016	Taylor Castonguay	Architectural Engineering	Demo building identification, fault test design
	Binbin Fan	Mechanical Engineering	Fault test implementation
2017	Wenjian Yu	Mechanical Engineering	Fault test implementation
	Benjamin Scheinberg	Mechanical Engineering	Fault test evaluation
	Mahamoudou Doumbia	Electrical Engineering	Fault energy impact analysis
	Taiyu Chen	Computer Engineering	AFDD tool interface development
	Ojas Pradhan	Mechanical Engineering	AFDD method evaluation
	Noelle Wiggins	Business	AFDD tool marketplace analysis
	Jeremy Solomon	Business	AFDD tool marketplace analysis
2018	Ojas Pradhan	Mechanical Engineering	AFDD method evaluation
	Taiyu Chen	Computer Engineering	AFDD tool interface development
	Crompton Hans	Architectural Engineering	BAS data collection
	Jaymes Bailey	Architectural Engineering	Fault test evaluation
	Noor Solatch	Mechanical Engineering	AFDD method evaluation

Chapter 11 Conclusions and Recommendations

11.1 Conclusions

In this report, the development and evaluation of the developed VOLTTRON™ compatible whole building AFDD tools are discussed. Different from component level AFDD solutions, the developed whole building AFDD tools aim at detecting, diagnosing and isolating whole building faults, i.e., those faults that occur in one component/subsystem, but have impacts on different components/subsystems, or have significant impacts on whole building performance. Main achievements from this project include:

- Developed a fault detection method that employs weather/schedule based pattern matching and feature engineering based principal component analysis techniques to detect whole building faults.
- Developed a fault diagnosis method that employs weather/schedule based pattern matching and BN technique to integrate expert knowledge and machine learning techniques to isolate the root causes of whole building faults.
- Building operational data that contain fault behavior caused by both naturally occurred and artificially implemented faults are collected for multiple seasons from a medium-sized university campus building. A variety of whole building faults that affect both primary and secondary systems have been implemented in this demonstration building for multiple seasons.
- The ground truth for these collected real building data, i.e., the evaluation data set, is established by an extensive manual data analysis process. In test cases that contain multiple simultaneous faults, the ground truth for the root causes of those naturally occurred faults could not be verified due to the limitation of demonstration building accessibility.
- Using the collected evaluation data set, the WPM-FPCA fault detection method is evaluated under two different PC retention rates, i.e., 0.65 and 0.95. It is observed that when using 0.65 PC retention rate, WPM-FPCA fault detection method achieves a 79% fault detection rate (this rate reaches 85% when including the six makeup fault test cases), while when using 0.95 PC retention rate, WPM-FPCA fault detection method achieves 100% fault detection rate (this rate remains 100% when including the six makeup fault test cases). Under both PC retention rates, WPM-FPCA fault detection method demonstrates 0% false alarm rates.
- Using the evaluation data set, the developed WPM-BN fault diagnosis method has successfully diagnosed thirteen (13) faults out of the fourteen (14) fault test cases in which only single fault exists. The one mis-diagnosed case exhibits fewer fault symptoms in different subsystems, when compared with other fault test cases. In a fault day when there are simultaneous occurred faults, the accuracy of the WPM-BN fault diagnosis method could not be completely established due to a lack of ground truth about the root causes.

- Overall speaking, the developed whole building AFDD tools demonstrate satisfactory fault detection/diagnosis accuracy with very low false alarm rate.
- These developed methods are highly automated, computational efficient, and do not require special training with an estimated implementation cost of ~$0.12 per square foot. The simple payback time is less than 3 years.
- An estimated 5 - 40% of HVAC system energy can be saved by diagnosing and correcting the faults investigated.
- Twelve undergraduate students from different disciplines have been engaged and exposed to AFDD in the building industry. Educational materials, including an online demo site, have been developed that can be used to educate and train workforce in the FDD field.

11.2 Recommendations

This research demonstrates the feasibility of using data-driven whole building AFDD methods, which integrate artificial intelligence techniques with domain knowledge, to accurately and cost-effectively detect and diagnose faults that have a whole building level impact. However, further research are needed to evaluate the developed solutions in more real buildings and with more types of faults to examine the proposed solutions' performance and scalability. The following recommendations are summarized from the conduct of this project:

11.2.1 Recommendation on More Evaluation using Real Building Testbeds

Due to the limited budget and time of this project, only one real building testbed with a limited number of faults are employed to evaluate the developed tools. In order to evaluate more thoroughly the performance, and especially the scalability, of the developed solutions, more buildings and more whole building faults need to be engaged in the future. It is recommended also to be aware that many faults may not cause detectable fault impacts under certain weather/operation/internal load conditions. For example, an outdoor damper stuck fault would not have any adverse impact if the weather/internal conditions actually would result in an outdoor air damper position that is close to the position that the damper is stuck at. Moreover, other naturally occurred or design faults could complicate fault impacts expected from an artificially implemented fault. For example, a chilled water differential pressure sensor bias fault would not cause the variable frequency pump to behave differently from its baseline if the chilled water system is not sized correctly. Therefore, careful examination of a real building testbed is needed prior to the design of an artificial fault implementation. Enough time needs to be budgeted for real building testing to allow potential failures and complications.

In this study, energy impact has been the main focus when analyzing fault impacts and selecting key measurements for PCA modeling and abnormality detection. A whole building fault may cause adverse impacts on other building performances such as indoor air quality. More comprehensive fault impact evaluation should be made in future real building demonstration and evaluation studies.

11.2.2 Recommendation on Improvement of WPM-FPCA Fault Detection Method

For the WPM-FPCA fault detection method developed in this study, the following issues/improvements can be explored in the future:

1. Currently, a fixed snapshot window mechanism is used to generate the WPM baseline dataset. In this study, such mechanism has delivered satisfactory performance in terms of accuracy and computation efficiency. However, if in a building where operation mode varies frequently, such a fixed snapshot window mechanism could potentially cause a false alarm. Therefore, in the future, a sliding window mechanism can be explored.

2. In this study, outdoor air enthalpy is the only index that is used to indicate weather condition and schedule is the only index that is used to indicate internal load due to a lack of occupancy information in the demonstration building. If occupancy information is available in future studies, it should be explored to be combined with weather related index to more accurately generate a baseline dataset.

3. Advanced techniques, such as Kernel PCA, Independent Component Analysis (ICA), and Kernel ICA, have been reported in the literature to be able to overcome certain shortcomings of traditional PCA, including a lack of ability to handle non-Gaussian dataset. These advanced techniques can be examined in future studies to improve the efficacy of the WPM-FPCA method.

11.2.3 Recommendation of the WPM-BN Fault Diagnosis Method

For the WPM-BN fault diagnosis method developed in this study, the following issues/improvements can be explored in the future:

1. Due to a lack of information about building fault frequency and impact in general, and due to the diversity of building design and systems, it is very hard to obtain accurate prior probability distribution for each fault, as well as conditional probability distribution for each fault evidence. In this study, literature and expert knowledge are used to obtain prior probability and is set to be universal for all faults considered. At the same time, the conditional probability distributions are discretized into several categorized levels. Again, domain knowledge is used to generate these distribution levels. More fault frequency data should be collected to provide more insights on the probability values used in this study. More sensitivity tests should be conduct in the future to examine the robustness of the method.

2. As illustrated in Chapter 8, simultaneous multiple faults happen in a real building. Symptoms from different faults could affect each other. How to obtain ground truth of fault ranking when multiple faults exist needs to be explored. Although the WPM-BN method has demonstrated great potential to diagnose faults even when multiple faults exist, evaluating BN's output under the scenario of multiple faults remain to be challenging.

11.2.4 Recommendation on Baseline Data Collection

Data quantity and quality play a critical role for any data-driven method, including data-driven AFDD methods. For the demonstration building in this study, which has a data sample interval of 5 minutes, it is found that at least 30 days of baseline data need to be collected for each season.

How to more effectively collect baseline data is an interesting topic that worth exploring in future research.

References

[1] DOE, Buildings Energy Data Book, in: U.S.D.o. Energy (Ed.), Washington D.C, 2011.

[2] M.R. Brambley, P. Haves, S.C. McDonald, P.A. Torcellini, D. Hansen, D. Holmberg, K. Roth, Advanced sensors and controls for building applications: Market assessment and potential R & D pathways, Pacific Northwest National Laboratory Washington, DC, USA, 2005.

[3] M.R. Brambley, P. Haves, S.C. McDonald, P. Torcellini, D.G. Hansen, D. Holmberg, K. Roth, Advanced sensors and controls for building applications: Market assessment and potential R&D pathways, in, Pacific Northwest National Laboratory (PNNL), Richland, WA (US), 2005.

[4] K.W. Roth, D. Westphalen, P. Llana, M. Feng, The energy impact of faults in US commercial buildings, (2004).

[5] M. Liu, L. Song, D. Claridge, Development of Whole Building Fault Detection Methods. High Performance Commercial Building Systems, California Energy Commission, Public Interest Energy Research Program, (2001).

[6] S. Katipamula, M.R. Brambley, Review article: methods for fault detection, diagnostics, and prognostics for building systems—a review, Part I, HVAC&R Research, 11 (1) (2005) 3-25.

[7] S. Katipamula, N.N. Bauman, R.G. Pratt, M.R. Brambley, Demonstration of the Whole-Building Diagnostician in a Single-Building Operator Environment, Pacific Northwest National Laboratory, 2003.

[8] K.W. Roth, D. Westphalen, M.Y. Feng, P. Llana, L. Quartararo, Energy impact of commercial building controls and performance diagnostics: market characterization, energy impact of building faults and energy savings potential, Prepared by TAIX LLC for the US Department of Energy. November. 412pp (Table 2–1), (2005).

[9] S. Li, J. Wen, Development and Validation of a Dynamic Air Handling Unit Model, Part I, ASHRAE Transactions, 116 (1) (2010).

[10] S. Ginestet, D. Marchio, O. Morisot, Evaluation of faults impacts on energy consumption and indoor air quality on an air handling unit, Energy and Buildings, 40 (1) (2008) 51-57.

[11] S. Ginestet, D. Marchio, Control tuning of a simplified VAV system: Methodology and impact on energy consumption and IAQ, Energy and Buildings, 42 (8) (2010) 1205-1214.

[12] S. Li, J. Wen, X. Zhou, C.J. Klaassen, Development and validation of a dynamic air handling unit model, part 2, ASHRAE Trans, 116 (1) (2010) 57-73.

[13] G.J. Vachtsevanos, F. Lewis, A. Hess, B. Wu, Intelligent fault diagnosis and prognosis for engineering systems, Wiley Online Library, 2006.

[14] B. Cai, L. Huang, M. Xie, Bayesian Networks in Fault Diagnosis, IEEE Transactions on Industrial Informatics, (2017).

[15] P. Haves, Overview of diagnostic methods, in: Proceedings of the Workshop on Diagnostics for Commercial Buildings: From Research to Practice, 1999, pp. 16-17.

[16] S. Katipamula, M.R. Brambley, N. Bauman, R.G. Pratt, Enhancing building operations through automated diagnostics: Field test results, (2003).

[17] J. Santos, L. Brightbill, L. Lister, Automated diagnostics from DDC data–PACRAT, in: Proceedings of the 8 th National Conference on Building Commissioning, Kansas City, MO, 2000.

[18] M. Liu, Improving building energy system performance by continuous commissioning, Energy Engineering, 96 (5) (1999) 46-57.

[19] J. Hyvarinen, S. Karki, Building optimization and fault diagnosis source book, IEA Annex, 25 (1996.8) (1996).

[20] J.M. House, G.E. Kelly, An overview of building diagnostics, in: National Conference on Building Commissioning, Kansas City, MO, 2000.

[21] S. Wu, J.Q. Sun, A top-down strategy with temporal and spatial partition for fault detection and diagnosis of building HVAC systems, Energy and Buildings, 43 (9) (2011) 2134-2139.

[22] S. Wu, J.-Q. Sun, Cross-level fault detection and diagnosis of building HVAC systems, Building and Environment, 46 (8) (2011) 1558-1566.

[23] J.E. Seem, Using intelligent data analysis to detect abnormal energy consumption in buildings, Energy and Buildings, 39 (1) (2007) 52-58.

[24] R.H. Dodier, J.F. Kreider, Detecting whole building energy problems, ASHRAE transactions, 105 (1999) 579.

[25] G. Lin, D.E. Claridge, A temperature-based approach to detect abnormal building energy consumption, Energy and Buildings, 93 (2015) 110-118.

[26] Z. Du, B. Fan, X. Jin, J. Chi, Fault detection and diagnosis for buildings and HVAC systems using combined neural networks and subtractive clustering analysis, Building and Environment, 73 (2014) 1-11.

[27] S.U. Lee, F.L. Painter, D.E. Claridge, Whole-Building Commercial HVAC System Simulation for Use in Energy Consumption Fault Detection, Ashrae Transactions, 113 (2) (2007).

[28] S. Katipamula, M.R. Brambley, Methods for fault detection, diagnostics, and prognostics for building systems—a review, part I, Hvac&R Research, 11 (1) (2005) 3-25.

[29] S. Katipamula, M.R. Brambley, Methods for fault detection, diagnostics, and prognostics for building systems—a review, part II, Hvac&R Research, 11 (2) (2005) 169-187.

[30] V. Venkatasubramanian, R. Rengaswamy, S.N. Kavuri, A review of process fault detection and diagnosis: Part II: Qualitative models and search strategies, Computers & Chemical Engineering, 27 (3) (2003) 313-326.

[31] D.B. Crawley, C.O. Pedersen, L.K. Lawrie, F.C. Winkelmann, EnergyPlus: energy simulation program, ASHRAE journal, 42 (4) (2000) 49.

[32] D.R. Clark, W.B. May, HVACSIM+ building systems and equipment simulation program-user's guide, in, National Bureau of Standards, Washington, DC (USA). Building Equipment Div., 1985.

[33] A. Trnsys, Transient System Simulation Program, University of Wisconsin, (2000).

[34] P. Haves, T. Salsbury, D. Claridge, M. Liu, Use of whole building simulation in on-line performance assessment: Modeling and implementation issues, Lawrence Berkeley National Laboratory, (2001).

[35] J.D. Bynum, D.E. Claridge, J.M. Curtin, Development and testing of an Automated Building Commissioning Analysis Tool (ABCAT), Energy and Buildings, 55 (2012) 607-617.

[36] Z. O'Neill, X. Pang, M. Shashanka, P. Haves, T. Bailey, Model-based real-time whole building energy performance monitoring and diagnostics, Journal of Building Performance Simulation, 7 (2) (2014) 83-99.

[37] B. Dong, Z. O'Neill, Z. Li, A BIM-enabled information infrastructure for building energy Fault Detection and Diagnostics, Automation in Construction, 44 (2014) 197-211.

[38] S. Li, J. Wen, Application of pattern matching method for detecting faults in air handling unit system, Automation in Construction, 43 (2014) 49-58.

[39] Y. Zhao, S. Wang, F. Xiao, Pattern recognition-based chillers fault detection method using support vector data description (SVDD), Applied Energy, 112 (2013) 1041-1048.

[40] S.J. Qin, Survey on data-driven industrial process monitoring and diagnosis, Annual Reviews in Control, 36 (2) (2012) 220-234.

[41] C. Miller, Z. Nagy, A. Schlueter, Automated daily pattern filtering of measured building performance data, Automation in Construction, 49, Part A (2015) 1-17.

[42] B. Yu, D. van Paassen, Fuzzy neural networks model for building energy diagnosis, in: 8th International IBPSA Conference, 2003, pp. 1459-1466.

[43] X. Li, C.P. Bowers, T. Schnier, Classification of energy consumption in buildings with outlier detection, Industrial Electronics, IEEE Transactions on, 57 (11) (2010) 3639-3644.

[44] F. Magoulès, H.-x. Zhao, D. Elizondo, Development of an RDP neural network for building energy consumption fault detection and diagnosis, Energy and Buildings, 62 (2013) 133-138.

[45] D. Jacob, S. Dietz, S. Komhard, C. Neumann, S. Herkel, Black-box models for fault detection and performance monitoring of buildings, Journal of Building Performance Simulation, 3 (1) (2010) 53-62.

[46] S. Yin, S.X. Ding, X. Xie, H. Luo, A review on basic data-driven approaches for industrial process monitoring, Industrial Electronics, IEEE Transactions on, 61 (11) (2014) 6418-6428.

[47] C. Aghemo, L. Blaso, A. Pellegrino, Building automation and control systems: A case study to evaluate the energy and environmental performances of a lighting control system in offices, Automation in Construction, 43 (2014) 10-22.

[48] L.M. Blanes, A. Costa, N. Réhault, M.M. Keane, Integration of Fault Detection and Diagnosis with Energy Management Standard ISO 50001 and Operations and Maitenance of HVAC Systems, in.

[49] C. Fan, F. Xiao, H. Madsen, D. Wang, Temporal knowledge discovery in big BAS data for building energy management, Energy and Buildings, 109 (2015) 75-89.

[50] A. Capozzoli, F. Lauro, I. Khan, Fault detection analysis using data mining techniques for a cluster of smart office buildings, Expert Systems with Applications, 42 (9) (2015) 4324-4338.

[51] S. Li, J. Wen, A model-based fault detection and diagnostic methodology based on PCA method and wavelet transform, Energy and Buildings, 68 (2014) 63-71.

[52] R. Srinivasan, M. Qian, State-specific key variables for monitoring multi-state processes, Chemical Engineering Research and Design, 85 (12) (2007) 1630-1644.

[53] B. Chebel-Morello, S. Malinowski, H. Senoussi, Feature selection for fault detection systems: application to the Tennessee Eastman process, Applied Intelligence, 44 (1) (2016) 111-122.

[54] K. Ghosh, M. Ramteke, R. Srinivasan, Optimal variable selection for effective statistical process monitoring, Computers & Chemical Engineering, 60 (2014) 260-276.

[55] C.J. Burges, Dimension reduction: A guided tour, Now Publishers Inc, 2010.

[56] A.L. Blum, P. Langley, Selection of relevant features and examples in machine learning, Artificial intelligence, 97 (1) (1997) 245-271.

[57] P. Langley, Selection of relevant features in machine learning, in: Proceedings of the AAAI Fall symposium on relevance, 1994, pp. 245-271.

[58] H. Motoda, H. Liu, Feature selection, extraction and construction, Communication of IICM (Institute of Information and Computing Machinery, Taiwan) Vol, 5 (2002) 67-72.

[59] A. Jain, D. Zongker, Feature selection: Evaluation, application, and small sample performance, IEEE transactions on pattern analysis and machine intelligence, 19 (2) (1997) 153-158.

[60] Y. Saeys, I. Inza, P. Larrañaga, A review of feature selection techniques in bioinformatics, bioinformatics, 23 (19) (2007) 2507-2517.

[61] T. Mehmood, K.H. Liland, L. Snipen, S. Sæbø, A review of variable selection methods in partial least squares regression, Chemometrics and Intelligent Laboratory Systems, 118 (2012) 62-69.

[62] I. Guyon, A. Elisseeff, An introduction to variable and feature selection, Journal of machine learning research, 3 (Mar) (2003) 1157-1182.

[63] M. Barker, W. Rayens, Partial least squares for discrimination, Journal of chemometrics, 17 (3) (2003) 166-173.

[64] H. Abdi, Partial least squares regression and projection on latent structure regression (PLS Regression), Wiley Interdisciplinary Reviews: Computational Statistics, 2 (1) (2010) 97-106.

[65] S. Wold, M. Sjöström, L. Eriksson, PLS-regression: a basic tool of chemometrics, Chemometrics and intelligent laboratory systems, 58 (2) (2001) 109-130.

[66] K. Fujiwara, H. Sawada, M. Kano, Input variable selection for PLS modeling using nearest correlation spectral clustering, Chemometrics and Intelligent Laboratory Systems, 118 (2012) 109-119.

[67] R. Kohavi, G.H. John, Wrappers for feature subset selection, Artificial intelligence, 97 (1-2) (1997) 273-324.

[68] R. Leardi, A.L. Gonzalez, Genetic algorithms applied to feature selection in PLS regression: how and when to use them, Chemometrics and intelligent laboratory systems, 41 (2) (1998) 195-207.

[69] W. Cai, Y. Li, X. Shao, A variable selection method based on uninformative variable elimination for multivariate calibration of near-infrared spectra, Chemometrics and intelligent laboratory systems, 90 (2) (2008) 188-194.

[70] J.A.F. Pierna, O. Abbas, V. Baeten, P. Dardenne, A Backward Variable Selection method for PLS regression (BVSPLS), Analytica chimica acta, 642 (1) (2009) 89-93.

[71] H.-D. Li, M.-M. Zeng, B.-B. Tan, Y.-Z. Liang, Q.-S. Xu, D.-S. Cao, Recipe for revealing informative metabolites based on model population analysis, Metabolomics, 6 (3) (2010) 353-361.

[72] T. Murata, H. Ishibuchi, H. Tanaka, Multi-objective genetic algorithm and its applications to flowshop scheduling, Computers & Industrial Engineering, 30 (4) (1996) 957-968.

[73] M. Mitchell, An introduction to genetic algorithms, MIT press, 1998.

[74] W. Siedlecki, J. Sklansky, A note on genetic algorithms for large-scale feature selection, Pattern recognition letters, 10 (5) (1989) 335-347.

[75] K. Hasegawa, Y. Miyashita, K. Funatsu, GA strategy for variable selection in QSAR studies: GA-based PLS analysis of calcium channel antagonists⊥, Journal of Chemical Information and Computer Sciences, 37 (2) (1997) 306-310.

[76] C. Bergmeir, J.M. Benítez, On the use of cross-validation for time series predictor evaluation, Information Sciences, 191 (2012) 192-213.

[77] B.M. Wise, N. Gallagher, R. Bro, J. Shaver, W. Windig, R.S. Koch, PLS Toolbox 4.0, Eigenvector Research Incorporated, 3905 (2007).

[78] X. Zhang, Y. Li, F. Lv, Complex batch processes quality prediction using non-Gaussian dissimilarity measure based just-in-time learning model, IFAC-PapersOnLine, 48 (21) (2015) 595-600.

[79] A. Regnier, Automated Fault Diagnostics for AHU-VAV Systems: A Novel Pattern Matching Approach, Thesis Proposal, (2015).

[80] Y. Zhao, F. Xiao, J. Wen, Y. Lu, S. Wang, A robust pattern recognition-based fault detection and diagnosis (FDD) method for chillers, HVAC&R Research, 20 (7) (2014) 798-809.

[81] C. Cheng, M.-S. Chiu, A new data-based methodology for nonlinear process modeling, Chemical Engineering Science, 59 (13) (2004) 2801-2810.

[82] C. Ratanamahatana, E. Keogh, A.J. Bagnall, S. Lonardi, A novel bit level time series representation with implication of similarity search and clustering, in: Pacific-Asia Conference on Knowledge Discovery and Data Mining, Springer, 2005, pp. 771-777.

[83] E. Keogh, K. Chakrabarti, M. Pazzani, S. Mehrotra, Dimensionality reduction for fast similarity search in large time series databases, Knowledge and information Systems, 3 (3) (2001) 263-286.

[84] J. Lin, E. Keogh, S. Lonardi, Visualizing and discovering non-trivial patterns in large time series databases, Information visualization, 4 (2) (2005) 61-82.

[85] M. Kano, S. Hasebe, I. Hashimoto, H. Ohno, A new multivariate statistical process monitoring method using principal component analysis, Computers & chemical engineering, 25 (7) (2001) 1103-1113.

[86] L.H. Chiang, R.D. Braatz, E.L. Russell, Fault detection and diagnosis in industrial systems, Springer Science & Business Media, 2001.

[87] S. Wang, F. Xiao, AHU sensor fault diagnosis using principal component analysis method, Energy and Buildings, 36 (2) (2004) 147-160.

[88] S. Wang, J. Qin, Sensor fault detection and validation of VAV terminals in air conditioning systems, Energy Conversion and Management, 46 (15) (2005) 2482-2500.

[89] S. Wang, J. Cui, Sensor-fault detection, diagnosis and estimation for centrifugal chiller systems using principal-component analysis method, Applied Energy, 82 (3) (2005) 197-213.

[90] Y. Hu, H. Chen, J. Xie, X. Yang, C. Zhou, Chiller sensor fault detection using a self-adaptive principal component analysis method, Energy and buildings, 54 (2012) 252-258.

[91] Z. Du, X. Jin, L. Wu, Fault detection and diagnosis based on improved PCA with JAA method in VAV systems, Building and Environment, 42 (9) (2007) 3221-3232.

[92] X. Xu, F. Xiao, S. Wang, Enhanced chiller sensor fault detection, diagnosis and estimation using wavelet analysis and principal component analysis methods, Applied Thermal Engineering, 28 (2) (2008) 226-237.

[93] J. Lin, E. Keogh, S. Lonardi, B. Chiu, A symbolic representation of time series, with implications for streaming algorithms, in: Proceedings of the 8th ACM SIGMOD workshop on Research issues in data mining and knowledge discovery, ACM, 2003, pp. 2-11.

[94] I. Jolliffe, Principal component analysis, Wiley Online Library, 2002.

[95] J.E. Jackson, A user's guide to principal components, John Wiley & Sons, 2005.

[96] P. Ralston, G. DePuy, J.H. Graham, Computer-based monitoring and fault diagnosis: a chemical process case study, ISA Transactions, 40 (1) (2001) 85-98.

[97] M. Zhu, A. Ghodsi, Automatic dimensionality selection from the scree plot via the use of profile likelihood, Computational Statistics & Data Analysis, 51 (2) (2006) 918-930.

[98] E. Martin, A. Morris, M. Papazoglou, Confidence bounds for multivariate process performance monitoring charts, IFAC Proceedings Volumes, 28 (12) (1995) 23-32.

[99] M. Lampis, J. Andrews, Bayesian belief networks for system fault diagnostics, Quality and Reliability Engineering International, 25 (4) (2009) 409-426.

[100] Y. Zhao, F. Xiao, S. Wang, An intelligent chiller fault detection and diagnosis methodology using Bayesian belief network, Energy and Buildings, 57 (2013) 278-288.

[101] Y. Zhao, J. Wen, F. Xiao, X. Yang, S. Wang, Diagnostic Bayesian networks for diagnosing air handling units faults–part I: Faults in dampers, fans, filters and sensors, Applied Thermal Engineering, 111 (2017) 1272-1286.

[102] Y. Zhao, J. Wen, S. Wang, Diagnostic Bayesian networks for diagnosing air handling units faults–Part II: Faults in coils and sensors, Applied Thermal Engineering, 90 (2015) 145-157.

[103] F. Xiao, Y. Zhao, J. Wen, S. Wang, Bayesian network based FDD strategy for variable air volume terminals, Automation in construction, 41 (2014) 106-118.

[104] S. He, Z. Wang, Z. Wang, X. Gu, Z. Yan, Fault detection and diagnosis of chiller using Bayesian network classifier with probabilistic boundary, Applied Thermal Engineering, 107 (2016) 37-47.

[105] Y. Chen, J. Wen, A whole building fault detection using weather based pattern matching and feature based PCA method, in: Big Data (Big Data), 2017 IEEE International Conference on, IEEE, 2017, pp. 4050-4057.

[106] D. Barber, Bayesian reasoning and machine learning, Cambridge University Press, 2012.

[107] M.J. Drużdżel, Intelligent Decision Support Systems Based on SMILE.

[108] A. Zagorecki, M.J. Druzdzel, Knowledge engineering for Bayesian networks: How common are noisy-MAX distributions in practice?, IEEE Transactions on Systems, Man, and Cybernetics: Systems, 43 (1) (2013) 186-195.

[109] S.F. Galán, F.J. Díez, Modeling dynamic causal interaction with Bayesian networks: temporal noisy gates, in: Proc. 2nd Inter. Workshop on Causal Networks, 2000, pp. 1-5.

[110] A. Regnier, J. Wen, Automated Fault Diagnostics for AHU-VAV Systems: A Bayesian Network Approach, in: 4th High Performance Buildings, West Lafayette, Indiana, U.S., 2016.

[111] Y. Chen, J. Wen, A whole building fault detection using weather based pattern matching and feature based PCA method, in: 2017 IEEE International Conference on Big Data, IEEE, Boston, Massachusetts, U.S., 2017, pp. 4050-4057.

[112] M.F. Sanner, Python: a programming language for software integration and development, J Mol Graph Model, 17 (1) (1999) 57-61.

Appendix I Fault cases in Year 2016 and Year 2017

Table I-1 Fault case in Year 2016

Date	Day	Time		Whole Building Fault Description				
		Start	End	Fault Description	Implementation	Fault Type	Normal Operation	Fault Implementation Season
07/06/16	Wednesday	16:00	23:00	System stopped at 4:00PM to 23:30PM	Naturally occurred fault	Naturally occurred fault	Normally system is on	Summer
08/01/16	Monday	15:59	21:30	AHU-1 supply air pressure sensor negative bias 0.2 inH2O	Override SP setpt from 1.8 inH2O to 2.0 inH2O	Supply air faults	Normally AHU-1 SA DP is 1.8 inH2O	Summer
08/08/16	Monday	10:22	21:16	AHU-1 supply air temperature sensor negative bias 4°F	Override DEM adj by 2	Supply air faults	Normally AHU-1 SA temperature is 60 °F in summer	Summer
08/13/16	Saturday	6:30	8:00	Earlier occupied	Occupied from 8AM to 6:30AM	Operator/scheduling faults	Normally system is off	Summer
08/16/16	Tuesday	11:28	21:41	Chiller chilled water differential pressure sensor positive bias 0.5 Psi	Override DP setpoint to 8 Psi	Primary cooling system faults	Normally chilled water differential pressure is 5.5 Psi	Summer
08/18/16	Thursday	9:30	19:17	AHU-1 outdoor air damper stuck at 30% open (higher than normal)	Override AHU-1 OA damper control signal on 30% open	Supply air faults	Normal AHU-1 outdoor air damper opens at 15% open	Summer
08/30/16	Tuesday	11:20	19:00	Chiller chilled water differential pressure sensor positive bias 0.5 Psi	Override DP setpoint to 7 Psi	Primary cooling system faults	Normally chilled water differential pressure is 5.5 Psi	Summer
09/07/16	Wednesday	10:30	22:09	AHU-2 supply air temperature sensor negative bias 4°F	Override DEM adj by 2	Supply air faults	Normally AHU-2 SA temperature is 60 °F in summer	Summer
09/11/16	Sunday	18:30	20:30	Operator fault, chiller off	Turn off chiller from 6:30 PM to 8:30PM	Operator/scheduling faults	Normally chiller is on	Summer
09/12/16	Monday	11:02	21:01	AHU-1 supply air temperature sensor negative bias 6°F	Override DEM adj by 3	Supply air faults	Normally AHU-1 SA temperature is 60 °F in summer	Summer
10/18/16	Tuesday	12:05	22:00	Chiller chilled water supply temperature sensor negative bias 2°F	Manually insert the value to chiller control pannel +2°F	Primary cooling system faults	Normally chilled water supply temperature is 44 °F	Transition
10/19/16	Wednesday	10:48	21:06	Chiller chilled water supply temperature sensor negative bias 4°F	Manually insert the value to chiller control pannel +4°F	Primary cooling system faults	Normally chilled water supply temperature is 44 °F	Transition
11/18/16	Friday	10:00	20:32	AHU-1 outdoor air damper override at 70% open (higher than normal)	Override AHU-1 OA damper control signal on 70% open	Supply air faults	Normal AHU-1 outdoor air damper opens at 15% open	Transition
12/01/16	Thursday	10:00	20:45	AHU-2 outdoor air damper stuck at 90% open (higher than normal)	Override AHU-2 OA damper control signal on 90% open	Supply air faults	Normal AHU-2 outdoor air damper opens at 15% open	Transition

Table I-2 Fault case in Year 2017

Date	Day	Time		Whole Building Fault Description				
		Start	End	Fault Description	Implementation	Fault Type	Normal Operation	Fault Implementation Season
01/03/17	Monday	10:00	20:30	AHU-2 outdoor air damper stuck at 80% open (higher than normal)	Override AHU-2 OA damper control signal on 80% open	Supply air faults	Normal AHU-2 outdoor air damper opens at 15% open	Winter
01/14/17	Saturday	1:30	7:00	Occupied from 1:30 AM to 7:00 AM	Naturally occurred fault	Naturally occurred fault	Normally chiller is on	Winter
01/16/17	Monday	10:00	20:30	AHU-2 supply air temperature sensor negative bias 4°F	Override DEM adj by 2	Supply air faults	Normally AHU-2 SA temperature is 60 °F in summer	Winter
03/29/17	Wednesday	10:02	21:19	AHU-2 outdoor air damper stuck at 90% open (higher than normal)	AHU-2 OA Damper stuck on 90% open	Supply air faults	Normal AHU-2 outdoor air damper opens at 15% open	Transition
04/13/17	Thursday	10:20	21:04	AHU-2 supply air temperature sensor negative bias 4°F	Override DEM adj by 2	Supply air faults	Normally AHU-2 SA temperature is 65 °F in summer	Transition
06/17/17	Saturday	10:00	22:20	Chiller plant chilled water supply bypass software override at 25% open (higher than normal)	Override bypass valve 25%	Operator/scheduling faults	Normally chilled water supply bypass valve open at 5% position	Summer
06/30/17	Friday	10:00	21:45	Chiller plant chilled water bypass valve software override at 30% open (higher than normal)	Override bypass valve 30%	Operator/scheduling faults	Normally chilled water supply bypass valve open at 5% position	Summer
07/09/17	Sunday	4:00	15:30	Chiller stopped earlier	Naturally occurred fault	Naturally occurred fault	Normally system is on	Summer
07/11/17	Tuesday	10:00	20:01	AHU-2 outdoor air damper stuck at 90% open (higher than normal)	Override AHU-2 OA damper control signal on 90% open	Supply air faults	Normal AHU-2 outdoor air damper opens at 15% open	Summer
07/18/17	Tuesday	11:00	20:01	AHU-2 outdoor air damper stuck at 100% open (higher than normal)	Override AHU-2 OA damper control signal on 100% open	Supply air faults	Normal AHU-2 outdoor air damper opens at 15% open	Summer
07/22/17	Saturday	10:00	21:57	Chiller differential pressure sensor positive bias 0.2 psi	Override the DP setpoint from 7.5 psi to 1 psi	Primary cooling system faults	Normally chilled water differential pressure is 5.5 Psi	Summer
08/03/17	Thursday	10:00	21:27	Chiller chilled water supply temperature sensor negative bias 4°F	Override the CHWS setpoint from 44 °F to 48 °F on control panel	Primary cooling system faults	Normally chilled water supply temperature is 44 °F	Summer
08/05/17	Saturday	4:00	8:00	HVAC system earlier occupied	Occupied from 4:00 AM	Operator/scheduling faults	Normally system is off	Summer
08/11/17	Friday	10:05	20:06	AHU-2 cooling coil valve position software override at 100% open (higher than normal)	Override cooling coil valve from BAS	Operator/scheduling faults	Normally AHU-2 cooling coil valve opens at 60% position	Summer
08/23/17	Wednesday	10:55	20:04	Chiller chilled water supply temperature negative bias 6°F	Override the CHWS setpoint from 44 °F to 50 °F on control panel	Primary cooling system faults	Excluded because of failure of collecting BAS data from August 17 to 29	Summer
09/15/17	Friday	10:30	15:21	Chiller chilled water differential pressure sensor positive bias 0.1 psi	Override the DP setpoint from 7.5 to 0.5	Primary cooling system faults	Normally chilled water differential pressure is 7.5 Psi	Summer

Appendix II List of Whole Building AFDD Tool Code

The Whole Building AFDD methods are developed based on MATLAB® R2015b. In order to implement demo, MATLAB® should be installed firstly. The code list is provided as below

Table II-1 Code list of AFDD tools

Function Name	Description	Version Control	Latest Updated Date
Main function			
Whole_building_AFDD_testbed_SAX_Weather_V7.m	Main function	V7	2018/7/2
Test Configuration			
Gather_Test_Info_V1.m	Gather test information	V1	2018/1/2
Gather_Test_Parameter_Setting_V1.m	Gather test parameters	V1	2018/1/2
Nesbitt_HVAC_Equipment_Ix.m	Gather test building equipment information	V1	2018/5/2
Data Collection			
Data_collection_4.m	Load test data	V4	2017/1/17
test_period_conversion_V1.m	Convert the test time period	V1	2017/5/17
Historical_database_generation_V3.m	Collect historical fault free data	V3	2018/1/20
Data_ORG_2.m	Data clean and rearrange	V2	2017/1/17
SAX method			
SAX_PM_T1.m	Implement SAX method	V1	2017/1/16
Wheather_day_extract_Ext_V4.m	Group weather data	V4	2017/5/18
PCA method			
PCA_calculation_T1.m	Generate baseline PCA model	V1	2017/1/16
PCA_test_data_statistic.m	PCA statistic calculation	V1	2017/1/16

Table II-1 Code list of AFDD tools (Cont'd)

Function Name	Description	Version Control	Latest Updated Date
Detection post process			
plot_detection_result_2.m	Plot statistic chart	V2	2018/8/20
BN method			
BN_inference_V2.m	Implement BN inference	V2	2017/10/17
Gen_BN_from_Xls_Dynamic_V1.m	Generate BN network	V1	2017/5/10
BN_evidence_gen_V3.m	BN evidence generation	V3	2018/10/25
Call_BN_structure.m	Call BN structure model	V1	2017/10/17
LEAK_generate_auto_V2.m	Generate LEAK probability	V2	2018/1/17
Method evaluation			
Equipment_pattern_analysis_V3.m	Equipment operation analysis	V3	2017/11/11
Energy_consumption_analysis_V1.m	Energy consumption evaluation	V1	2018/3/22
BN_Posterior_Prob_Analsis_V1.m	BN posterior probability analysis and plot	V1	2018/3/1
weather_pattern_analysis_3.m	Weather pattern analysis	V3	2017/4/17

Appendix III List of VOLTTRON™ and BAS Interface Code

The VOLTTRON™ to BAS interface driver is developed under Python 2.7.3 version. In order to run the driver, Python 2.7 should be installed. Python third-party packages such as CSV, Suds, datatime, os, shutil and so on should be firstly imported. The list of the developed codes is provided as below.

Table III-1 Code list of VOLTTRON™ and BAS interface driver

Code List		
Module Name	**Description**	**Version Control**
send_to_csv.py	Send BAS data to a csv file	V1
date_time_org.py	Obtain time stamp from the BAS	V1
get_data.py	Collect BAS data	V1
Read_address.py	Read BAS data measurement address	V1
voltton_webctrl_V1.py	VOLTTRON to BAS interface driver	V1
call_webctrl_trend.py	Call BAS trend data paths	V1
splitcsv_Date.py	Split csv data file according to the date	V1

Appendix IV Market Survey Instrument

Two surveys were conducted to understand the market barriers and gaps: Service Providers Survey (https://drexel.qualtrics.com/SE/?SID=SV_6ujZWI0IPOJf4r3) and Building Owner/Operator Survey (https://drexel.qualtrics.com/SE/?SID=SV_78rH3vujFJ5PlKB). Screen shots of these two surveys are provided here to illustrate the details.

1) Service Providers Survey:

Thank you for agreeing to take part in this important survey measuring market trends for automatic fault detection and diagnostic software for Drexel University. Today, we will be gaining your thoughts and opinions in order to better serve you in the future.
This survey will only take 3-4 minutes to complete. Be assured that all answers you provide will be kept in the strictest confidentiality. *Your email address will be stored only to track survey completion. The data will be reported only in the aggregate and no individual information will be identified.*

Contact Details

Name	
Email	
Phone Number	
Company/ Organization	
Position	

What is automated fault detection and diagnostic software?

Automated Fault Detection and Diagnostic Software (AFDD): An automated process through which a fault (such as a malfunctioned sensor or equipment) is detected (I.e., system considered abnormal) and then the actual cause of the fault is isolated.

Do you currently provide fault diagnostics as part of your software offerings?

- Yes
- No

What types of entities are your typical customers? (eg. individual buildings, third-party service providers, etc.)

Have you considered improvements to your current software offerings in the past year?

- ○ Yes
- ○ No

Which of the following characteristics are most important to you in diagnostic software?

- ☐ Low Cost
- ☐ Diagnostic Accuracy/ Low False Alarm Rates
- ☐ Data Visualization
- ☐ Easy Installation
- ☐ Other (please specify)

Rank the following diagnostic software characteristics as most important (1) and least important (3). Drag and drop the numbers into order.

- Low Cost of Implementation (Economy)
- Comprehensive Fault Detection (Sensitivity)
- Few False Alarms (Specificity)

Approximately what percentage of your clients use your diagnostic software package?

- ○ Do not offer
- ○ Less than 10%
- ○ 10%-29%
- ○ 30%-49%
- ○ 50%-79%
- ○ More than 80%

Which 3 of the following do you feel are the most significant barriers to larger market penetration of diagnostic software?

- [ ] Lack of Customer Education
- [ ] Lack of Consumer Interest
- [ ] Lack of Knowledge of Existence of Software
- [ ] Lack of Benchmarking Standards
- [ ] Lack of Performance Standards
- [ ] Existing Products Don't Work Well Enough
- [ ] Existing Products are Too Hard to Install
- [ ] Existing Products are Too Expensive
- [ ] Difficult to Evaluate Performance
- [ ] Not Required in Government Coding Standards
- [ ] Other (please specify)

Estimate the percentage of all commercial buildings currently using fault diagnostic software.

0 10 20 30 40 50 60 70 80 90 100

Percentage

Based on the diagnostic software you use, estimate the cost for the consumer in a "typical" medium-sized (50,000 square foot) commercial building.

Upfront, set up, or one-time purchasing cost per square foot:	
(per square foot cost, optional)	
Ongoing cost of monitoring per square foot per year:	

Do you think that the current cost of diagnostic software acts as a major barrier for consumer adoption of diagnostic services?

- ( ) Definitely yes
- ( ) Probably yes
- ( ) Probably not

How much do you think the cost of diagnostic software must be lowered to minimize the barrier to adoption?

- Less than 10%
- 10-29%
- 30-49%
- 50-80%
- Greater than 80%

What is the price your consumers are willing to pay for diagnostic software?

Upfront, set up, or one-time purchasing cost:	
(per square foot cost, optional)	
Ongoing cost of monitoring per square foot per year:	

How do you anticipate prices in diagnostic software will change in the next five years?

- Increase
- No Change
- Decrease

<< >>

Thank you for taking this survey. Your insight and professional opinion is helping research at Drexel University!

Is there anything else you think we should know about the market or fault diagnostic software?

<< >>

2) Building Owner/Operator Survey:

Thank you for agreeing to take part in this important survey measuring market trends for automatic fault detection and diagnostic software for Drexel University. Today, we will be gaining your thoughts and opinions in order to better serve you in the future with the software we create.
This survey will only take 3-4 minutes to complete. Be assured that all answers you provide will be kept in the strictest confidentiality. *Your email address will be stored only to track survey completion. The data will be reported only in the aggregate and no individual will be identified.*

Please give us a little information!

Name	
Email	
Phone Number	
Company/Organization	
Position	

Which of the following building types do you manage and/or own?

- ☐ Offices
- ☐ Warehouses
- ☐ Residential Properties
- ☐ Community Buildings
- ☐ Religious
- ☐ Government Buildings
- ☐ Entertainment
- ☐ Sports Facilities
- ☐ Hotels and Motels
- ☐ Educational Facilities
- ☐ Medical
- ☐ Retail
- ☐ Restaurants
- ☐ Transportation
- ☐ Power Station
- ☐ Data Center
- ☐ Other (please specify) _____

How long ago did you purchase your current diagnostic software?

- ○ Less Than 6 Months
- ○ 6 months- 1 year
- ○ 1-2 years
- ○ 2 or more years

Since you last bought your diagnostic software, have you researched upgrading or buying new diagnostic software?

- ○ Yes
- ○ No

In order, what were the three most important factors when choosing a diagnostic software? (Drag and Drop 1 for most important, 5 for least important)

- Energy/Cost Savings
- Improved Indoor Air Quality
- Improved Occupant Comfort
- Improved Maintenance/Repair Scheduling
- Overall Easier Management of Building

How satisfied are you with your current diagnostic software?

Not at all Satisfied										Extremely Satisfied
0	1	2	3	4	5	6	7	8	9	10
○	○	○	○	○	○	○	○	○	○	○

What is a reasonable cost for diagnostic software in one of your buildings?

Upfront, set up, or one-time purchasing cost:

(per square foot cost, optional)

Ongoing cost of monitoring per square foot per year:

How likely are you to buy the software for you buildings at that price?

- Extremely likely
- Moderately likely
- Slightly likely
- Neither likely nor unlikely
- Slightly unlikely
- Moderately unlikely
- Extremely unlikely

When have you last researched with intent to buy diagnostic software?

- 6 months
- 1 year
- 2 years
- More than 2 years
- Never

Why did you choose not to buy diagnostic software?

- Haven't researched it/Didn't know it was available
- Software Not Reliable
- Previous Bad Experience
- Too Expensive
- Difficult to Install
- Too Complicated to Use
- Other (please specify)

How many of the following features are most important to you when looking at diagnostic software?

- ☐ Diagnostic Accuracy/ Low False Alarm Rate
- ☐ Lowered Energy Costs
- ☐ Integration with Service Scheduling Software
- ☐ Data Visualization
- ☐ Easy Installation
- ☐ Low Cost

Which pricing structure do you prefer?

- ○ High Per Month Price
- ○ High up front cost and low per month price
- ○ Very high up front cost, no monthly rate

Thank you so much for taking the time to complete our survey. Your insight is helping us develop new technology to better serve your needs!

Is there anything else you think we should know for the purposes of our research?

Appendix V Whole Building Fault Energy Impacts

Fault Type	Date	Day	Time		Whole Building Fault Description		Fault Impacts on Energy Consumption	
			Start	End	Fault Description	Implementation	Whole Building Electricity	CHW cooling load
OpF-Sch-Unocc	07/06/16	Wednesday	16:00	23:00	System stopped at 4:00PM to 23:30PM	Naturally occurred fault	5.6%higher than baseline	99.7%lower than baseline
	07/22/17	Saturday	20:20	21:40	System stopped at 20:20 to 21:40	Manually change schedule to unoccupied	4.4%higher than baseline	98.2%lower than baseline
OpF-Chiller-Off	09/11/16	Sunday	18:30	20:30	Operator fault, chiller off	Turn off chiller from 6:30 PM to 8:30PM	0.7%lower than baseline	34%lower than baseline
	07/09/17	Sunday	4:00	15:30	Chiller stopped earlier	Naturally occurred fault	11.4%lower than baseline	100%lower than baseline
OpF-Sch-Occ	01/14/17	Saturday	1:30	7:00	Occupied from 1:30 AM to 7:00 AM	Naturally occurred fault	3.2%higher than baseline	N/A
	08/05/17	Saturday	4:00	8:00	HVAC system earlier occupied	Occupied from 4:00 AM	0.8%higher than baseline	114%higher than baseline
OpF-CC-VLV-SWO-H	08/11/17	Friday	10:05	20:06	AHU-2 cooling coil valve position software override fault (higher than normal)	Override Cooling coil valve from BAS	2.4%higher than baseline	21.8%higher than baseline
CHWS-DP-Bias-P	09/15/17	Friday	10:30	15:21	Chiller chilled water differential pressure sensor positive bias 0.1 psi	Override the DP setpoint from 7.5 to 0.5	0.7%lower than baseline	93%lower than baseline
CHWS-Temp-Bias-N	08/03/17	Thursday	10:00	21:27	Chiller chilled water supply temperature sensor negative bias 4°F	Override the CHWS setpoint from 44 °F to 48 °F on control panel	0.2%higher than baseline	2.8%higher than baseline
	07/23/18	Sunday	8:00	18:00	Chiller chilled water supply temperature sensor negative bias 3°F	Override the CHWS setpoint from 44 °F to 47 °F on control panel	1.3%lower than baseline	0.9%higher than baseline
AHU-SA-Temp-Bias-N	08/08/16	Monday	10:22	21:16	AHU-1 supply air temperature sensor negative bias 4°F	Override DEM adj by 2	1.8%lower than baseline	15.6%lower than baseline
	09/07/16	Wednesday	10:30	22:09	AHU-2 supply air temperature sensor negative bias 4°F	Override DEM adj by 2	3.6%higher than baseline	4.9%lower than baseline
	07/09/18	Monday	10:30	22:09	AHU-2 supply air temperature sensor bias fault negative 3.5°F	Override DEM adj by 1.75	0.6%lower than baseline	3.5%lower than baseline
AHU-OA-DMPR-Stuck-H	12/01/16	Thursday	10:00	20:45	AHU-2 outdoor air damper stuck at 90% open (higher than normal)	Override AHU-2 OA damper at 90%open	1.6%higher than baseline	N/A
	01/03/17	Monday	10:00	20:30	AHU-2 outdoor air damper stuck at 80% open (higher than normal)	Override AHU-2 OA damper at 80%open	5.9%higher than baseline	N/A
	07/11/17	Tuesday	10:00	20:01	AHU-2 outdoor air damper stuck at 90% open (higher than normal)	AHU-2 OA damper stuck on 90%naturally occurred in AHU-1	10%higher than baseline	39.3%higher than baseline
	07/18/17	Tuesday	11:00	20:01	AHU-2 outdoor air damper stuck at 100%open (higher than normal)	AHU-2 OA damper stuck on 100%	3.8%higher than baseline	28.4%higher than baseline
	07/10/18	Tuesday	10:30	20:30	AHU-2 OA damper stuck at 30%open (higher than normal)	AHU-2 OA damper stuck on 30%	0.5%lower than baseline	3.5%higher than baseline
	07/18/18	Wednesday	9:30	19:30	AHU-2 OA damper stuck at 60%open (higher than normal)	AHU-2 OA damper stuck on 60%	1.1%higher than baseline	6.1%higher than baseline
AHU-CC-VLV-Stuck-H	07/11/18	Thursday	10:00	20:10	AHU-2 cooling coil valve stuck at 80% (higher than the normal position)	Override cooling coil valve control signal at 80% open, record the valve driver control signal	3.3%higher than baseline	45%higher than baseline

Appendix VI Variable Selection Result

Table VI-1 Variable Selection Result (summer season)

Variable No.	Component	Measurement Name	Selection	Variable No.	Component	Measurement Name	Selection	Variable No.	Component	Measurement Name	Selection
1	OA conditions	Outdoor air temperature	√	30	AHU-1	Sensor: Return air temperature	√	59		Sensor: frozen status	√
2		Outdoor air relative humidity	√	31		Sensor: Mixed air temperature	√	60		Sensor: Between-coil temperature	√
3		Outdoor air enthalpy	√	32		Sensor: Supply air temperature	√	61		Sensor: Supply air static pressure	√
4	PECO meter	Energy Usage Hour KWh	x	33		Control: Supply fan speed	x	62		Supply air temperature setpoint	√
5	Chiller	Sensor: Chilled water flowrate	√	34		Economizer position	√	63		Supply air static pressure setpoint	√
6		Sensor: Chilled water return temperature	√	35		Feedback: Cooling coil valve position	√	64		Sensor: Return air relative humidity	√
7		Sensor: Chilled water supply temperature	√	36		Economizer setpoint	√	65		Feedback: Supply air fan speed	x
8		Sensor: Cooling water delta pressure	x	37		Control: Return fan speed	√	66		Feedback: Return air fan speed	x
9		Sensor: Cooling water return temperature	√	38		Sensor: Filter DP	x	67		Sensor: Supply air humidity	√
10		Sensor: Cooling water supply temperature	√	39		Sensor: Between-coil temperature	√	68		Sensor: Supply air fan current	√
11		Sensor: Cooling water valve feedback	√	40		Sensor: Supply air static pressure	√	69		Sensor: Return air fan current	√
12		Sensor: Chilled water supply delta pressure	√	41		Supply air temperature setpoint	x	70		Control: Steam valve 1/3	x
13		Control chiller status	x	42		Supply air static pressure setpoint	x	71		Control: Steam valve 2/3	x
14		Spare pump status	x	43		Sensor: Return air relative humidity	x	72		Control: Relief Air damper	√
15		Bypass valve status	x	44		Feedback: Supply air fan speed	x	73		Control: Return Air damper	√
16		Control: Chilled water bypass valve output	√	45		Feedback: Return air fan speed	x	74	AHU-3	Sensor: Supply air humidity	√
17		Control: Chilled water pump vfd output	x	46		Sensor: Supply air fan current	√	75		Sensor: Supply air temperature	x
18		Control: Cooling water pump vfd output	√	47		Sensor: Return air fan current	√	76		Sf_vfd_output	x
19		Control: Chiller on/off	x	48		Sensor: frozen status	x	77		Economizer position	x
20		Control: Chilled water pump on/off	x	49		Control: Steam valve 1/3	x	78		Feedback: Cooling coil valve position	√
21		Control: Cooling water pump on/off	x	50		Control: Steam valve 2/3	x	79		Sensor: frozen status	x
22		Control: Spare pump on/off	x	50	AHU-2	Sensor: Return air temperature	√	80		Sensor: Between-coil temperature	x
23		Control: Cooling water bypass setpoint	x	52		Sensor: Mixed air temperature	√	81		Sensor: Supply air static pressure	x
24		Chilled water flowrate setpoint	x	53		Sensor: Supply air temperature	√	82		Supply air temperature setpoint	x
25		Bypass valve setpoint	x	54		Supply fan speed	√	83		Supply air temperature setpoint	x
26		Cooling water delta pressure setpoint	x	55		Economizer position	x	84		Sensor: Supply air humidity	√
27		Sensor: Cooling water delta pressure	x	56		Feedback: Cooling coil valve position	x	85		Sensor: Supply air fan current	√
28		Chilled water delta pressure setpoint	x	57		Economizer setpoint	√	86		Steam valve 1/3	x
29		Chilled water flowrate setpoint	x	58		Control Return fan speed	√	87		Steam valve 2/3	x

Table VI-1 Variable Selection Result (summer season) (Cont'd)

Variable No.	Component	Measurement Name	Selection	Variable No.	Component	Measurement Name	Selection	Variable No.	Component	Measurement Name	Selection
88	Cooling tower	CT Fan Status	x	116	VV-1-1-3	Sensor: Air flow rate	x	146	VV-1-1-8	Sensor: Air flow rate	x
89		valve Pos	x	117		Sensor: Zone temperature	x	147		Sensor: Zone temperature	√
90		CT Fan S/S	x	118		Sensor: Discharge temper	√	148		Sensor: Discharge temperature	√
91		CT VFD Stpt	x	119		Control:HW valve	√	149		Control:HW valve	x
92	VV-1-B1	Sensor: Flowrate	√	120		Control: Damper Position	√	150		Control: Damper Position	√
93		Sensor: Discharge Temperature	√	121		Control: Flow Stpt	√	151		Control: Flow Stpt	√
94		Control:HW Valve	x	122	VV-1-1-4	Sensor: Air flow rate	√	152	VV-1-1-9	Sensor: Air flow rate	√
95		Control: Damper Position	x	123		Sensor: Zone temperature	√	153		Sensor: Zone temperature	√
96		Control: Flow Stpt	x	124		Sensor: Discharge temper	√	154		Sensor: Discharge temperature	√
97		Cooling setpoint	x	125		Control:HW valve	x	155		Control:HW valve	x
98		Heating setpoint	x	126		Control: Damper Position	√	156		Control: Damper Position	√
99	VV-1-B-2	Sensor: Air flow rate	x	127		Control: Flow Stpt	√	157		Control: Flow Stpt	√
100		Sensor: Zone temperature	x	128	VV-1-1-5	Sensor: Air flow rate	√	158	VV-1-1-10	Sensor: Air flow rate	√
101		Sensor: Discharge temperature	x	129		Sensor: Zone temperature	√	159		Sensor: Zone temperature	√
102		Control: Damper Position	√	130		Sensor: Discharge temper	√	160		Sensor: Discharge temperature	√
103		Control: Flow Stpt	x	131		Control:HW valve	x	161		Control:HW valve	x
104	VV-1-1-1	Sensor: Air flow rate	x	132		Control: Damper Position	√	162		Control: Damper Position	√
105		Sensor: Zone temperature	x	133		Control: Flow Stpt	√	163		Control: Flow Stpt	√
106		Sensor: Discharge temperature	x	134	VV-1-1-6	Sensor: Air flow rate	x	164	VV-1-1-11	Sensor: Air flow rate	x
107		Control:HW valve	x	135		Sensor: Zone temperature	x	165		Sensor: Zone temperature	x
108		Control: Damper Position	x	136		Sensor: Discharge temper	x	166		Sensor: Discharge temperature	x
109		Control: Flow Stpt		137		Control:HW valve	x	167		Control:HW valve	x
110	VV-1-1-2	Sensor: Air flow rate	√	138		Control: Damper Position	x	168		Control: Damper Position	x
111		Sensor: Zone temperature	√	139		Control: Flow Stpt	x	169		Control: Flow Stpt	x
112		Sensor: Discharge temperature	√	140	VV-1-1-7	Sensor: Air flow rate	x	170	VV-1-1-12	Sensor: Air flow rate	x
113		Control:HW valve	x	141		Sensor: Zone temperature	√	171		Sensor: Zone temperature	x
114		Control: Damper Position	x	142		Sensor: Discharge temper	√	172		Sensor: Discharge temperature	x
115		Control: Flow Stpt	x	143		Control:HW valve	x	173		Control:HW valve	x
				144		Control: Damper Position	√	174		Control: Damper Position	x
				145		Control: Flow Stpt	√	175		Control: Flow Stpt	x

Table VI-1 Variable Selection Result (summer season) (Cont'd)

Variable No.	Component	Measurement Name	Selection	Variable No.	Component	Measurement Name	Selection	Variable No.	Component	Measurement Name	Selection
176	VV-2-2-1	Sensor: Air flow rate	×	212	VV-2-2-7	Sensor: Air flow rate	×	248	VV-2-3-3	Sensor: Air flow rate	×
177		Sensor: Zone temperature	√	213		Sensor: Zone temperature	×	249		Sensor: Zone temperature	√
178		Sensor: Discharge temperature	√	214		Sensor: Discharge temperat	×	250		Sensor: Discharge temperature	√
179		Control:HW valve	×	215		Control:HW valve	×	251		Control:HW valve	×
180		Control: Damper Position	×	216		Control: Damper Position	√	252		Control: Damper Position	×
181		Control: Flow Stpt	×	217		Control: Flow Stpt	√	253		Control: Flow Stpt	×
182	VV-2-2-2	Sensor: Air flow rate	×	218	VV-2-2-8	Sensor: Air flow rate	×	254	VV-2-3-4	Sensor: Air flow rate	×
183		Sensor: Zone temperature	√	219		Sensor: Zone temperature	×	255		Sensor: Zone temperature	×
184		Sensor: Discharge temperature	×	220		Sensor: Discharge temperat	×	256		Sensor: Discharge temperature	×
185		Control:HW valve	×	221		Control:HW valve	√	257		Control:HW valve	×
186		Control: Damper Position	√	222		Control: Damper Position	√	258		Control: Damper Position	√
187		Control: Flow Stpt	×	223		Control: Flow Stpt	×	259		Control: Flow Stpt	√
188	VV-2-2-3	Sensor: Air flow rate	×	224	VV-2-2-9	Sensor: Air flow rate	×	260	VV-2-3-5	Sensor: Air flow rate	×
189		Sensor: Zone temperature	×	225		Sensor: Zone temperature	×	261		Sensor: Zone temperature	√
190		Sensor: Discharge temperature	×	226		Sensor: Discharge temperat	×	262		Sensor: Discharge temperature	√
191		Control:HW valve	×	227		Control:HW valve	×	263		Control:HW valve	×
192		Control: Damper Position	√	228		Control: Damper Position	√	264		Control: Damper Position	×
193		Control: Flow Stpt	√	229		Control: Flow Stpt	√	265		Control: Flow Stpt	×
194	VV-2-2-4	Sensor: Air flow rate	×	230	VV-2-2-7	Sensor: Air flow rate	×	266	VV-2-3-6	Sensor: Air flow rate	×
195		Sensor: Zone temperature	×	231		Sensor: Zone temperature	×	267		Sensor: Zone temperature	×
196		Sensor: Discharge temperature	×	232		Sensor: Discharge temperat	×	268		Sensor: Discharge temperature	×
197		Control:HW valve	×	233		Control:HW valve	×	269		Control:HW valve	×
198		Control: Damper Position	×	234		Control: Damper Position	×	270		Control: Damper Position	√
199		Control: Flow Stpt	×	235		Control: Flow Stpt	×	271		Control: Flow Stpt	×
200	VV-2-2-5	Sensor: Air flow rate	×	236	VV-2-3-1	Sensor: Air flow rate	√	272	VV-2-3-7	Sensor: Air flow rate	×
201		Sensor: Zone temperature	×	237		Sensor: Zone temperature	√	273		Sensor: Zone temperature	×
202		Sensor: Discharge temperature	√	238		Sensor: Discharge temperat	×	274		Sensor: Discharge temperature	×
203		Control:HW valve	×	239		Control:HW valve	×	275		Control:HW valve	×
204		Control: Damper Position	√	240		Control: Damper Position	×	276		Control: Damper Position	√
205		Control: Flow Stpt	√	241		Control: Flow Stpt	×	277		Control: Flow Stpt	×
206	VV-2-2-6	Sensor: Air flow rate	×	242	VV-2-3-2	Sensor: Air flow rate	×	278	VV-2-3-8	Sensor: Air flow rate	×
207		Sensor: Zone temperature	×	243		Sensor: Zone temperature	×	279		Sensor: Zone temperature	×
208		Sensor: Discharge temperature	×	244		Sensor: Discharge temperat	×	280		Sensor: Discharge temperature	×
209		Control:HW valve	×	245		Control:HW valve	×	281		Control:HW valve	×
210		Control: Damper Position	√	246		Control: Damper Position	×	282		Control: Damper Position	×
211		Control: Flow Stpt	×	247		Control: Flow Stpt	×	283		Control: Flow Stpt	×

Table VI-1 Variable Selection Result (summer season) (Cont'd)

Variable No.	Component	Measurement Name	Selection	Variable No.	Component	Measurement Name	Selection	Variable No.	Component	Measurement Name	Selection
284	VV-2-3-9	Sensor: Air flow rate	x	320	VV-2-4-5	Sensor: Air flow rate	x	356	VV-2-5-1	Sensor: Air flow rate	x
285		Sensor: Zone temperature	x	321		Sensor: Zone temperature	√	357		Sensor: Zone temperature	x
286		Sensor: Discharge temperature	x	322		Sensor: Discharge temperat	√	358		Sensor: Discharge temperature	√
287		Control:HW valve	x	323		Control:HW valve	x	359		Control:HW valve	x
288		Control: Damper Position	x	324		Control: Damper Position	x	360		Control: Damper Position	x
289		Control: Flow Stpt	x	325		Control: Flow Stpt	x	361		Control: Flow Stpt	√
290	VV-2-3-10	Sensor: Air flow rate	√	326	VV-2-4-6	Sensor: Air flow rate	x	362	VV-2-5-2	Sensor: Air flow rate	x
291		Sensor: Zone temperature	√	327		Sensor: Zone temperature	x	363		Sensor: Zone temperature	x
292		Sensor: Discharge temperature	x	328		Sensor: Discharge temperat	x	364		Sensor: Discharge temperature	x
293		Control:HW valve	x	329		Control:HW valve	x	365		Control:HW valve	x
294		Control: Damper Position	√	330		Control: Damper Position	x	366		Control: Damper Position	x
295		Control: Flow Stpt	x	331		Control: Flow Stpt	x	367		Control: Flow Stpt	x
296	VV-2-4-1	Sensor: Air flow rate	x	332	VV-2-4-7	Sensor: Air flow rate	√	368	VV-2-5-3	Sensor: Air flow rate	√
297		Sensor: Zone temperature	x	333		Sensor: Zone temperature	x	369		Sensor: Zone temperature	√
298		Sensor: Discharge temperature	x	334		Sensor: Discharge temperat	x	370		Sensor: Discharge temperature	x
299		Control:HW valve	x	335		Control:HW valve	x	371		Control:HW valve	x
300		Control: Damper Position	x	336		Control: Damper Position	x	372		Control: Damper Position	√
301		Control: Flow Stpt	x	337		Control: Flow Stpt	x	373		Control: Flow Stpt	x
302	VV-2-4-2	Sensor: Air flow rate	x	338	VV-2-4-8	Sensor: Air flow rate	x	374	VV-2-5-4	Sensor: Air flow rate	x
303		Sensor: Zone temperature	x	339		Sensor: Zone temperature	x	375		Sensor: Zone temperature	x
304		Sensor: Discharge temperature	x	340		Sensor: Discharge temperat	x	376		Sensor: Discharge temperature	x
305		Control:HW valve	x	341		Control:HW valve	x	377		Control:HW valve	x
306		Control: Damper Position	x	342		Control: Damper Position	x	378		Control: Damper Position	x
307		Control: Flow Stpt	x	343		Control: Flow Stpt	x	379		Control: Flow Stpt	x
308	VV-2-4-3	Sensor: Air flow rate	x	344	VV-2-4-9	Sensor: Air flow rate	x	380	VV-2-5-5	Sensor: Air flow rate	x
309		Sensor: Zone temperature	x	345		Sensor: Zone temperature	x	381		Sensor: Zone temperature	x
310		Sensor: Discharge temperature	x	346		Sensor: Discharge temperat	√	382		Sensor: Discharge temperature	x
311		Control:HW valve	x	347		Control:HW valve	x	383		Control:HW valve	x
312		Control: Damper Position	x	348		Control: Damper Position	x	384		Control: Damper Position	√
313		Control: Flow Stpt	x	349		Control: Flow Stpt	x	385		Control: Flow Stpt	x
314	VV-2-4-4	Sensor: Air flow rate	x	350	VV-2-4-10	Sensor: Air flow rate	x	386	VV-2-5-6	Sensor: Air flow rate	x
315		Sensor: Zone temperature	x	351		Sensor: Zone temperature	x	387		Sensor: Zone temperature	x
316		Sensor: Discharge temperature	√	352		Sensor: Discharge temperat	x	388		Sensor: Discharge temperature	x
317		Control:HW valve	x	353		Control:HW valve	x	389		Control:HW valve	x
318		Control: Damper Position	x	354		Control: Damper Position	x	390		Control: Damper Position	x
319		Control: Flow Stpt	x	355		Control: Flow Stpt	x	391		Control: Flow Stpt	x

Table VI-1 Variable Selection Result (summer season) (Cont'd)

Variable No.	Component	Measurement Name	Selection	Variable No.	Component	Measurement Name	Selection	Variable No.	Component	Measurement Name	Selection
392	VV-2-5-7	Sensor: Air flow rate	√	422	VV-2-6-2	Sensor: Air flow rate	x	452	VV-2-6-7	Sensor: Air flow rate	x
393		Sensor: Zone temperature	√	423		Sensor: Zone temperature	x	453		Sensor: Zone temperature	x
394		Sensor: Discharge temperature	x	424		Sensor: Discharge temperature	x	454		Sensor: Discharge temperature	x
395		Control:HW valve	x	425		Control:HW valve	x	455		Control:HW valve	x
396		Control: Damper Position	x	426		Control: Damper Position	x	456		Control: Damper Position	x
397		Control: Flow Stpt	x	427		Control: Flow Stpt	x	457		Control: Flow Stpt	x
398	VV-2-5-8	Sensor: Air flow rate	x	428	VV-2-6-3	Sensor: Air flow rate	x	458	VV-2-6-8	Sensor: Air flow rate	√
399		Sensor: Zone temperature	x	429		Sensor: Zone temperature	x	459		Sensor: Zone temperature	√
400		Sensor: Discharge temperature	x	430		Sensor: Discharge temperature	x	460		Sensor: Discharge temperature	x
401		Control:HW valve	x	431		Control:HW valve	x	461		Control:HW valve	x
402		Control: Damper Position	x	432		Control: Damper Position	x	462		Control: Damper Position	x
403		Control: Flow Stpt	x	433		Control: Flow Stpt	x	463		Control: Flow Stpt	x
404	VV-2-5-9	Sensor: Air flow rate	x	434	VV-2-6-4	Sensor: Air flow rate	√	464	VV-2-6-9	Sensor: Air flow rate	x
405		Sensor: Zone temperature	x	435		Sensor: Zone temperature	√	465		Sensor: Zone temperature	x
406		Sensor: Discharge temperature	x	436		Sensor: Discharge temperature	x	466		Sensor: Discharge temperature	x
407		Control:HW valve	x	437		Control:HW valve	x	467		Control:HW valve	x
408		Control: Damper Position	x	438		Control: Damper Position	x	468		Control: Damper Position	x
409		Control: Flow Stpt	x	439		Control: Flow Stpt	x	469		Control: Flow Stpt	x
410	VV-2-5-10	Sensor: Air flow rate	x	440	VV-2-6-5	Sensor: Air flow rate	x	470	VV-2-6-10	Sensor: Air flow rate	x
411		Sensor: Zone temperature	x	441		Sensor: Zone temperature	x	471		Sensor: Zone temperature	x
412		Sensor: Discharge temperature	x	442		Sensor: Discharge temperature	x	472		Sensor: Discharge temperature	x
413		Control:HW valve	x	443		Control:HW valve	x	473		Control:HW valve	x
414		Control: Damper Position	x	444		Control: Damper Position	x	474		Control: Damper Position	√
415		Control: Flow Stpt	x	445		Control: Flow Stpt	x	475		Control: Flow Stpt	√
416	VV-2-6-1	Sensor: Air flow rate	√	446	VV-2-6-6	Sensor: Air flow rate	x	476	VV-2-7-1	Sensor: Air flow rate	x
417		Sensor: Zone temperature	x	447		Sensor: Zone temperature	x	477		Sensor: Zone temperature	x
418		Sensor: Discharge temperature	√	448		Sensor: Discharge temperature	√	478		Sensor: Discharge temperature	√
419		Control:HW valve	x	449		Control:HW valve	x	479		Control:HW valve	x
420		Control: Damper Position	x	450		Control: Damper Position	x	480		Control: Damper Position	x
421		Control: Flow Stpt	x	451		Control: Flow Stpt	x	481		Control: Flow Stpt	x

Table VI-1 Variable Selection Result (summer season) (Cont'd)

Variable No.	Component	Measurement Name	Selection	Variable No.	Component	Measurement Name	Selection
482	VV-2-7-2	Sensor: Air flow rate	×	512	VV-2-7-7	Sensor: Air flow rate	×
483		Sensor: Zone temperature	×	513		Sensor: Zone temperature	√
484		Sensor: Discharge temperature	×	514		Sensor: Discharge temperature	√
485		Control:HW valve	×	515		Control:HW valve	×
486		Control: Damper Position	×	516		Control: Damper Position	×
487		Control: Flow Stpt	×	517		Control: Flow Stpt	×
488	VV-2-7-3	Sensor: Air flow rate	×	518	VV-2-7-8	Sensor: Air flow rate	×
489		Sensor: Zone temperature	×	519		Sensor: Zone temperature	√
490		Sensor: Discharge temperature	×	520		Sensor: Discharge temperature	×
491		Control:HW valve	×	521		Control:HW valve	×
492		Control: Damper Position	×	522		Control: Damper Position	×
493		Control: Flow Stpt	×	523		Control: Flow Stpt	×
494	VV-2-7-4	Sensor: Air flow rate	×	524	VV-2-7-9	Sensor: Air flow rate	×
495		Sensor: Zone temperature	×	525		Sensor: Zone temperature	×
496		Sensor: Discharge temperature	×	526		Sensor: Discharge temperature	×
497		Control:HW valve	×	527		Control:HW valve	×
498		Control: Damper Position	×	528		Control: Damper Position	×
499		Control: Flow Stpt	×	529		Control: Flow Stpt	×
500	VV-2-7-5	Sensor: Air flow rate	×	530	Hx-1	Sensor: heat exchange1 supply water temperature	√
501		Sensor: Zone temperature	×	531		Sensor: Heat exchange 1 return water tempearture	√
502		Sensor: Discharge temperature	×	532	Hx-2	Sensor: Heat exchange 2 supply water tempearture1	√
503		Control:HW valve	×	533		Sensor: Heat exchange 2 return water tempearture1	×
504		Control: Damper Position	×	534		Sensor: Heat exchange 2 supply water tempearture2	×
505		Control: Flow Stpt	×	535		Sensor: Heat exchange 2 return water tempearture2	×
506	VV-2-7-6	Sensor: Air flow rate	×	536		Control: Heat exchange 2 1/3 valve openness	×
507		Sensor: Zone temperature	×	537		Control: Heat exchange 2 2/3 valve openness	×
508		Sensor: Discharge temperature	×				
509		Control:HW valve	×				
510		Control: Damper Position	×				
511		Control: Flow Stpt	×				

Table VI-2 Variable Selection Result (winter and transition season)

Variable No.	Component	Measurement Name	Selection	Variable No.	Component	Measurement Name	Selection	Variable No.	Component	Measurement Name	Selection
1	OA conditions	Outdoor air temperature	√	30	AHU-1	Sensor: Return air temperature	√	59		Sensor: frozen status	√
2		Outdoor air relative humidity	√	31		Sensor: Mixed air temperature	√	60		Sensor: Between-coil temperature	√
3		Outdoor air enthalpy	√	32		Sensor: Supply air temperature	√	61		Sensor: Supply air static pressure	√
4	PECO meter	Energy Usage Hour KWh	x	33		Control: Supply fan speed	x	62		Supply air temperature setpoint	√
5	Chiller	Sensor: Chilled water flowrate	x	34		Economizer position	√	63		Supply air static pressure setpoint	√
6		Sensor: Chilled water return temperature	x	35		Feedback: Cooling coil valve position	√	64		Sensor: Return air relative humidity	√
7		Sensor: Chilled water supply temperature	x	36		Economizer setpoint	√	65		Feedback: Supply air fan speed	x
8		Sensor: Cooling water delta pressure	x	37		Control: Return fan speed	x	66		Feedback: Return air fan speed	x
9		Sensor: Cooling water return temperature	x	38		Sensor: Filter DP	x	67		Sensor: Supply air humidity	√
10		Sensor: Cooling water supply temperature	x	39		Sensor: Between-coil temperature	√	68		Sensor: Supply air fan current	√
11		Sensor: Cooling water valve feedback	x	40		Sensor: Supply air static pressure	x	69		Sensor: Return air fan current	x
12		Sensor: Chilled water supply delta pressure	x	41		Supply air temperature setpoint	x	70		Control: Steam valve 1/3	√
13		Control chiller status	x	42		Supply air static pressure setpoint	x	71		Control: Steam valve 2/3	x
14		Spare pump status	x	43		Sensor: Return air relative humidity	x	72		Control: Relief Air damper	√
15		Bypass valve status	x	44		Feedback: Supply air fan speed	x	73		Control: Return Air damper	√
16		Control: Chilled water bypass valve output	x	45		Feedback: Return air fan speed	x	74	AHU-3	Sensor: Supply air humidity	√
17		Control: Chilled water pump vfd output	x	46		Sensor: Supply air fan current	√	75		Sensor: Supply air temperature	x
18		Control: Cooling water pump vfd output	x	47		Sensor: Return air fan current	x	76		Sf_vfd_output	x
19		Control: Chiller on/off	x	48		Sensor: frozen status	x	77		Economizer position	x
20		Control: Chilled water pump on/off	x	49		Control: Steam valve 1/3	√	78		Feedback: Cooling coil valve position	√
21		Control: Cooling water pump on/off	x	50		Control: Steam valve 2/3	x	79		Sensor: frozen status	x
22		Control: Spare pump on/off	x	50	AHU-2	Sensor: Return air temperature	√	80		Sensor: Between-coil temperature	x
23		Control: Cooling water bypass setpoint	x	52		Sensor: Mixed air temperature	√	81		Sensor: Supply air static pressure	x
24		Chilled water flowrate setpoint	x	53		Sensor: Supply air temperature	√	82		Supply air temperature setpoint	x
25		Bypass valve setpoint	x	54		Supply fan speed	√	83		Supply air temperature setpoint	x
26		Cooling water delta pressure setpoint	x	55		Economizer position	x	84		Sensor: Supply air humidity	√
27		Sensor: Cooling water delta pressure	x	56		Feedback: Cooling coil valve position	x	85		Sensor: Supply air fan current	√
28		Chilled water delta pressure setpoint	x	57		Economizer setpoint	√	86		Steam valve 1/3	x
29		Chilled water flowrate setpoint	x	58		Control Return fan speed	√	87		Steam valve 2/3	x

Table VI-2 Variable Selection Result (winter and transition season) (Cont'd)

Variable No.	Component	Measurement Name	Selection	Variable No.	Component	Measurement Name	Selection	Variable No.	Component	Measurement Name	Selection
88	Cooling tower	CT Fan Status	x	116	VV-1-1-3	Sensor: Air flow rate	x	146	VV-1-1-8	Sensor: Air flow rate	x
89		valve Pos	x	117		Sensor: Zone temperature	x	147		Sensor: Zone temperature	√
90		CT Fan S/S	x	118		Sensor: Discharge temperature	√	148		Sensor: Discharge temperature	√
91		CT VFD Stpt	x	119		Control:HW valve	√	149		Control:HW valve	x
92	VV-1-B1	Sensor: Flowrate	√	120		Control: Damper Position	√	150		Control: Damper Position	√
93		Sensor: Discharge Temperature	√	121		Control: Flow Stpt	√	151		Control: Flow Stpt	√
94		Control:HW Valve	x	122	VV-1-1-4	Sensor: Air flow rate	√	152	VV-1-1-9	Sensor: Air flow rate	√
95		Control: Damper Position	x	123		Sensor: Zone temperature	√	153		Sensor: Zone temperature	√
96		Control: Flow Stpt	x	124		Sensor: Discharge temperature	√	154		Sensor: Discharge temperature	√
97		Cooling setpoint	x	125		Control:HW valve	x	155		Control:HW valve	x
98		Heating setpoint	x	126		Control: Damper Position	√	156		Control: Damper Position	√
99	VV-1-B-2	Sensor: Air flow rate	x	127		Control: Flow Stpt	√	157		Control: Flow Stpt	√
100		Sensor: Zone temperature	x	128	VV-1-1-5	Sensor: Air flow rate	√	158	VV-1-1-10	Sensor: Air flow rate	√
101		Sensor: Discharge temperature	x	129		Sensor: Zone temperature	√	159		Sensor: Zone temperature	√
102		Control: Damper Position	√	130		Sensor: Discharge temperature	√	160		Sensor: Discharge temperature	√
103		Control: Flow Stpt	x	131		Control:HW valve	x	161		Control:HW valve	x
104	VV-1-1-1	Sensor: Air flow rate	x	132		Control: Damper Position	√	162		Control: Damper Position	√
105		Sensor: Zone temperature	x	133		Control: Flow Stpt	√	163		Control: Flow Stpt	√
106		Sensor: Discharge temperature	x	134	VV-1-1-6	Sensor: Air flow rate	x	164	VV-1-1-11	Sensor: Air flow rate	x
107		Control:HW valve	x	135		Sensor: Zone temperature	x	165		Sensor: Zone temperature	x
108		Control: Damper Position	x	136		Sensor: Discharge temperature	x	166		Sensor: Discharge temperature	x
109		Control: Flow Stpt		137		Control:HW valve	x	167		Control:HW valve	x
110	VV-1-1-2	Sensor: Air flow rate	√	138		Control: Damper Position	x	168		Control: Damper Position	x
111		Sensor: Zone temperature	√	139		Control: Flow Stpt	x	169		Control: Flow Stpt	x
112		Sensor: Discharge temperature	√	140	VV-1-1-7	Sensor: Air flow rate	x	170	VV-1-1-12	Sensor: Air flow rate	x
113		Control:HW valve	x	141		Sensor: Zone temperature	√	171		Sensor: Zone temperature	x
114		Control: Damper Position	x	142		Sensor: Discharge temperature	√	172		Sensor: Discharge temperature	x
115		Control: Flow Stpt	x	143		Control:HW valve	x	173		Control:HW valve	x
				144		Control: Damper Position	√	174		Control: Damper Position	x
				145		Control: Flow Stpt	√	175		Control: Flow Stpt	x

Table VI-2 Variable Selection Result (winter and transition season) (Cont'd)

Variable No.	Component	Measurement Name	Selection	Variable No.	Component	Measurement Name	Selection	Variable No.	Component	Measurement Name	Selection
176	VV-2-2-1	Sensor: Air flow rate	x	212	VV-2-2-7	Sensor: Air flow rate	x	248	VV-2-3-3	Sensor: Air flow rate	x
177		Sensor: Zone temperature	√	213		Sensor: Zone temperature	x	249		Sensor: Zone temperature	√
178		Sensor: Discharge temperature	√	214		Sensor: Discharge temperature	x	250		Sensor: Discharge temperature	√
179		Control:HW valve	√	215		Control:HW valve	x	251		Control:HW valve	√
180		Control: Damper Position	x	216		Control: Damper Position	√	252		Control: Damper Position	x
181		Control: Flow Stpt	x	217		Control: Flow Stpt	√	253		Control: Flow Stpt	x
182	VV-2-2-2	Sensor: Air flow rate	x	218	VV-2-2-8	Sensor: Air flow rate	x	254	VV-2-3-4	Sensor: Air flow rate	x
183		Sensor: Zone temperature	√	219		Sensor: Zone temperature	x	255		Sensor: Zone temperature	x
184		Sensor: Discharge temperature	x	220		Sensor: Discharge temperature	x	256		Sensor: Discharge temperature	x
185		Control:HW valve	x	221		Control:HW valve	√	257		Control:HW valve	x
186		Control: Damper Position	√	222		Control: Damper Position	√	258		Control: Damper Position	√
187		Control: Flow Stpt	x	223		Control: Flow Stpt	x	259		Control: Flow Stpt	√
188	VV-2-2-3	Sensor: Air flow rate	x	224	VV-2-2-9	Sensor: Air flow rate	√	260	VV-2-3-5	Sensor: Air flow rate	x
189		Sensor: Zone temperature	x	225		Sensor: Zone temperature	x	261		Sensor: Zone temperature	√
190		Sensor: Discharge temperature	x	226		Sensor: Discharge temperature	x	262		Sensor: Discharge temperature	√
191		Control:HW valve	√	227		Control:HW valve	x	263		Control:HW valve	x
192		Control: Damper Position	√	228		Control: Damper Position	√	264		Control: Damper Position	x
193		Control: Flow Stpt	√	229		Control: Flow Stpt	√	265		Control: Flow Stpt	x
194	VV-2-2-4	Sensor: Air flow rate	x	230	VV-2-2-7	Sensor: Air flow rate	x	266	VV-2-3-6	Sensor: Air flow rate	x
195		Sensor: Zone temperature	x	231		Sensor: Zone temperature	x	267		Sensor: Zone temperature	x
196		Sensor: Discharge temperature	x	232		Sensor: Discharge temperature	x	268		Sensor: Discharge temperature	x
197		Control:HW valve	x	233		Control:HW valve	x	269		Control:HW valve	x
198		Control: Damper Position	x	234		Control: Damper Position	x	270		Control: Damper Position	√
199		Control: Flow Stpt	x	235		Control: Flow Stpt	x	271		Control: Flow Stpt	x
200	VV-2-2-5	Sensor: Air flow rate	√	236	VV-2-3-1	Sensor: Air flow rate	√	272	VV-2-3-7	Sensor: Air flow rate	x
201		Sensor: Zone temperature	x	237		Sensor: Zone temperature	√	273		Sensor: Zone temperature	x
202		Sensor: Discharge temperature	√	238		Sensor: Discharge temperature	x	274		Sensor: Discharge temperature	x
203		Control:HW valve	x	239		Control:HW valve	x	275		Control:HW valve	x
204		Control: Damper Position	√	240		Control: Damper Position	x	276		Control: Damper Position	√
205		Control: Flow Stpt	√	241		Control: Flow Stpt	x	277		Control: Flow Stpt	x
206	VV-2-2-6	Sensor: Air flow rate	x	242	VV-2-3-2	Sensor: Air flow rate	x	278	VV-2-3-8	Sensor: Air flow rate	x
207		Sensor: Zone temperature	x	243		Sensor: Zone temperature	x	279		Sensor: Zone temperature	x
208		Sensor: Discharge temperature	x	244		Sensor: Discharge temperature	x	280		Sensor: Discharge temperature	x
209		Control:HW valve	x	245		Control:HW valve	x	281		Control:HW valve	x
210		Control: Damper Position	√	246		Control: Damper Position	x	282		Control: Damper Position	x
211		Control: Flow Stpt	x	247		Control: Flow Stpt	x	283		Control: Flow Stpt	x

Table VI-2 Variable Selection Result (winter and transition season) (Cont'd)

Variable No.	Component	Measurement Name	Selection	Variable No.	Component	Measurement Name	Selection	Variable No.	Component	Measurement Name	Selection
284	VV-2-3-9	Sensor: Air flow rate	x	320	VV-2-4-5	Sensor: Air flow rate	x	356	VV-2-5-1	Sensor: Air flow rate	x
285		Sensor: Zone temperature	x	321		Sensor: Zone temperature	√	357		Sensor: Zone temperature	x
286		Sensor: Discharge temperature	x	322		Sensor: Discharge temperature	√	358		Sensor: Discharge temperature	√
287		Control:HW valve	x	323		Control:HW valve	x	359		Control:HW valve	x
288		Control: Damper Position	x	324		Control: Damper Position	x	360		Control: Damper Position	x
289		Control: Flow Stpt	x	325		Control: Flow Stpt	x	361		Control: Flow Stpt	√
290	VV-2-3-10	Sensor: Air flow rate	√	326	VV-2-4-6	Sensor: Air flow rate	√	362	VV-2-5-2	Sensor: Air flow rate	x
291		Sensor: Zone temperature	√	327		Sensor: Zone temperature	x	363		Sensor: Zone temperature	x
292		Sensor: Discharge temperature	x	328		Sensor: Discharge temperature	x	364		Sensor: Discharge temperature	x
293		Control:HW valve	x	329		Control:HW valve	√	365		Control:HW valve	x
294		Control: Damper Position	√	330		Control: Damper Position	x	366		Control: Damper Position	x
295		Control: Flow Stpt	x	331		Control: Flow Stpt	x	367		Control: Flow Stpt	x
296	VV-2-4-1	Sensor: Air flow rate	x	332	VV-2-4-7	Sensor: Air flow rate	√	368	VV-2-5-3	Sensor: Air flow rate	√
297		Sensor: Zone temperature	x	333		Sensor: Zone temperature	x	369		Sensor: Zone temperature	√
298		Sensor: Discharge temperature	x	334		Sensor: Discharge temperature	x	370		Sensor: Discharge temperature	x
299		Control:HW valve	x	335		Control:HW valve	x	371		Control:HW valve	x
300		Control: Damper Position	x	336		Control: Damper Position	x	372		Control: Damper Position	√
301		Control: Flow Stpt	x	337		Control: Flow Stpt	x	373		Control: Flow Stpt	x
302	VV-2-4-2	Sensor: Air flow rate	x	338	VV-2-4-8	Sensor: Air flow rate	x	374	VV-2-5-4	Sensor: Air flow rate	x
303		Sensor: Zone temperature	x	339		Sensor: Zone temperature	x	375		Sensor: Zone temperature	x
304		Sensor: Discharge temperature	x	340		Sensor: Discharge temperature	x	376		Sensor: Discharge temperature	x
305		Control:HW valve	√	341		Control:HW valve	x	377		Control:HW valve	x
306		Control: Damper Position	x	342		Control: Damper Position	x	378		Control: Damper Position	x
307		Control: Flow Stpt	x	343		Control: Flow Stpt	x	379		Control: Flow Stpt	x
308	VV-2-4-3	Sensor: Air flow rate	√	344	VV-2-4-9	Sensor: Air flow rate	x	380	VV-2-5-5	Sensor: Air flow rate	x
309		Sensor: Zone temperature	√	345		Sensor: Zone temperature	x	381		Sensor: Zone temperature	x
310		Sensor: Discharge temperature	x	346		Sensor: Discharge temperature	√	382		Sensor: Discharge temperature	x
311		Control:HW valve	√	347		Control:HW valve	x	383		Control:HW valve	x
312		Control: Damper Position	x	348		Control: Damper Position	x	384		Control: Damper Position	√
313		Control: Flow Stpt	x	349		Control: Flow Stpt	x	385		Control: Flow Stpt	x
314	VV-2-4-4	Sensor: Air flow rate	x	350	VV-2-4-10	Sensor: Air flow rate	x	386	VV-2-5-6	Sensor: Air flow rate	x
315		Sensor: Zone temperature	x	351		Sensor: Zone temperature	x	387		Sensor: Zone temperature	x
316		Sensor: Discharge temperature	√	352		Sensor: Discharge temperature	x	388		Sensor: Discharge temperature	x
317		Control:HW valve	x	353		Control:HW valve	x	389		Control:HW valve	x
318		Control: Damper Position	x	354		Control: Damper Position	x	390		Control: Damper Position	x
319		Control: Flow Stpt	x	355		Control: Flow Stpt	x	391		Control: Flow Stpt	x

Table VI-2 Variable Selection Result (winter and transition season) (Cont'd)

Variable No.	Component	Measurement Name	Selection	Variable No.	Component	Measurement Name	Selection	Variable No.	Component	Measurement Name	Selection
392	VV-2-5-7	Sensor: Air flow rate	√	422	VV-2-6-2	Sensor: Air flow rate	x	452	VV-2-6-7	Sensor: Air flow rate	x
393		Sensor: Zone temperature	√	423		Sensor: Zone temperature	x	453		Sensor: Zone temperature	x
394		Sensor: Discharge temperature	x	424		Sensor: Discharge temperature	x	454		Sensor: Discharge temperature	x
395		Control:HW valve	x	425		Control:HW valve	x	455		Control:HW valve	x
396		Control: Damper Position	x	426		Control: Damper Position	x	456		Control: Damper Position	x
397		Control: Flow Stpt	x	427		Control: Flow Stpt	x	457		Control: Flow Stpt	x
398	VV-2-5-8	Sensor: Air flow rate	x	428	VV-2-6-3	Sensor: Air flow rate	x	458	VV-2-6-8	Sensor: Air flow rate	√
399		Sensor: Zone temperature	x	429		Sensor: Zone temperature	x	459		Sensor: Zone temperature	√
400		Sensor: Discharge temperature	x	430		Sensor: Discharge temperature	x	460		Sensor: Discharge temperature	x
401		Control:HW valve	x	431		Control:HW valve	x	461		Control:HW valve	x
402		Control: Damper Position	x	432		Control: Damper Position	x	462		Control: Damper Position	x
403		Control: Flow Stpt	x	433		Control: Flow Stpt	x	463		Control: Flow Stpt	x
404	VV-2-5-9	Sensor: Air flow rate	x	434	VV-2-6-4	Sensor: Air flow rate	√	464	VV-2-6-9	Sensor: Air flow rate	x
405		Sensor: Zone temperature	x	435		Sensor: Zone temperature	√	465		Sensor: Zone temperature	x
406		Sensor: Discharge temperature	x	436		Sensor: Discharge temperature	x	466		Sensor: Discharge temperature	x
407		Control:HW valve	x	437		Control:HW valve	√	467		Control:HW valve	x
408		Control: Damper Position	x	438		Control: Damper Position	x	468		Control: Damper Position	x
409		Control: Flow Stpt	x	439		Control: Flow Stpt	x	469		Control: Flow Stpt	x
410	VV-2-5-10	Sensor: Air flow rate	x	440	VV-2-6-5	Sensor: Air flow rate	x	470	VV-2-6-10	Sensor: Air flow rate	x
411		Sensor: Zone temperature	x	441		Sensor: Zone temperature	x	471		Sensor: Zone temperature	x
412		Sensor: Discharge temperature	x	442		Sensor: Discharge temperature	x	472		Sensor: Discharge temperature	x
413		Control:HW valve	x	443		Control:HW valve	x	473		Control:HW valve	x
414		Control: Damper Position	x	444		Control: Damper Position	x	474		Control: Damper Position	√
415		Control: Flow Stpt	x	445		Control: Flow Stpt	x	475		Control: Flow Stpt	√
416	VV-2-6-1	Sensor: Air flow rate	√	446	VV-2-6-6	Sensor: Air flow rate	x	476	VV-2-7-1	Sensor: Air flow rate	x
417		Sensor: Zone temperature	x	447		Sensor: Zone temperature	x	477		Sensor: Zone temperature	√
418		Sensor: Discharge temperature	√	448		Sensor: Discharge temperature	√	478		Sensor: Discharge temperature	√
419		Control:HW valve	x	449		Control:HW valve	x	479		Control:HW valve	√
420		Control: Damper Position	x	450		Control: Damper Position	x	480		Control: Damper Position	x
421		Control: Flow Stpt	x	451		Control: Flow Stpt	x	481		Control: Flow Stpt	x

Table VI-2 Variable Selection Result (winter and transition season) (Cont'd)

Variable No.	Component	Measurement Name	Selection	Variable No.	Component	Measurement Name	Selection
482	VV-2-7-2	Sensor: Air flow rate	×	512	VV-2-7-7	Sensor: Air flow rate	×
483		Sensor: Zone temperature	×	513		Sensor: Zone temperature	√
484		Sensor: Discharge temperature	×	514		Sensor: Discharge temperature	√
485		Control:HW valve	×	515		Control:HW valve	×
486		Control: Damper Position	×	516		Control: Damper Position	×
487		Control: Flow Stpt	×	517		Control: Flow Stpt	×
488	VV-2-7-3	Sensor: Air flow rate	×	518	VV-2-7-8	Sensor: Air flow rate	×
489		Sensor: Zone temperature	√	519		Sensor: Zone temperature	√
490		Sensor: Discharge temperature	×	520		Sensor: Discharge temperature	×
491		Control:HW valve	×	521		Control:HW valve	×
492		Control: Damper Position	×	522		Control: Damper Position	×
493		Control: Flow Stpt	×	523		Control: Flow Stpt	×
494	VV-2-7-4	Sensor: Air flow rate	×	524	VV-2-7-9	Sensor: Air flow rate	×
495		Sensor: Zone temperature	×	525		Sensor: Zone temperature	×
496		Sensor: Discharge temperature	×	526		Sensor: Discharge temperature	×
497		Control:HW valve	√	527		Control:HW valve	×
498		Control: Damper Position	×	528		Control: Damper Position	×
499		Control: Flow Stpt	×	529		Control: Flow Stpt	×
500	VV-2-7-5	Sensor: Air flow rate	×	530	Hx-1	Sensor: heat exchange1 supply water temperature	√
501		Sensor: Zone temperature	×	531		Sensor: Heat exchange 1 return water tempearture	√
502		Sensor: Discharge temperature	×	532	Hx-2	Sensor: Heat exchange 2 supply water tempearture1	×
503		Control:HW valve	×	533		Sensor: Heat exchange 2 return water tempearture1	×
504		Control: Damper Position	×	534		Sensor: Heat exchange 2 supply water tempearture2	√
505		Control: Flow Stpt	×	535		Sensor: Heat exchange 2 return water tempearture2	×
506	VV-2-7-6	Sensor: Air flow rate	×	536		Control: Heat exchange 2 1/3 valve openness	√
507		Sensor: Zone temperature	√	537		Control: Heat exchange 2 2/3 valve openness	×
508		Sensor: Discharge temperature	√				
509		Control:HW valve	√				
510		Control: Damper Position	×				
511		Control: Flow Stpt	×				