Fault detection and diagnosis of the low ΔT syndrome in cooling coils of chilled water systems

Please check the document version of this publication:

• A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher's website.
• The final author version and the galley proof are versions of the publication after peer review.
• The final published version features the final layout of the paper including the volume, issue and page numbers.

[Link to publication](#)

Fault detection and diagnosis of the low ΔT syndrome in cooling coils of chilled water systems

Anand Thamban
29th August 2022

EINDHOVEN UNIVERSITY OF TECHNOLOGY

Stan Ackermans Institute

SMART BUILDINGS & CITIES

FAULT DETECTION AND DIAGNOSIS OF THE LOW ΔT SYNDROME IN COOLING COILS OF CHILLED WATER SYSTEMS

By

Anand Thamban

A thesis submitted in partial fulfilment of the requirements for the degree of
Professional Doctorate of Engineering

The design described in this thesis has been carried out in accordance with the TU/e Code of
Scientific Conduct

Prof. ir. Wim Zeiler, university coach

Ir. Alet van den Brink, company coach

Dr. ir. Shalika Walker, daily university coach

Eindhoven, the Netherlands

August, 2022

This thesis has been established in collaboration with

and project partners

A catalogue record is available from the Eindhoven University of Technology Library

Acknowledgements

During the two years of my PDEng traineeship, I was able to grow and develop my skills to become the skilled professional I am now. This was possible due to the opportunities provided to me by the different organisations involved in this project, and the support and guidance provided to me by my mentors.

Firstly, I would like to thank all the stakeholders in the project consortium, especially TKI Urban Energy and Eindhoven Energy who have supported the project. A special thanks to Kropman Installatietechniek for providing me with the opportunity to work on this project. I am very thankful to many members of the team for their support throughout. Jan-Willem Dubbeldam, Joris de Ruiter and Francesco Favi from TCC have been very helpful in providing their guidance in data collection and application development. The experiments for this study would not have been possible without the continuous support of John Verlaan, Roel Vos and Marco van As, who have always helped round the clock to ensure that the experiments were conducted properly.

During my time at TU/e, I received lots of support from my coaches and colleagues from the SBC program. I am grateful for their support throughout my project. I would also like to thank my colleagues at the Building Services group, who helped me in certain aspects of my project and provided me with a good work environment.

The greatest improvement in my skills and knowledge were only possible due to my mentors, and for this I am extremely grateful. I would like to thank Shalika for constantly supporting and guiding me in my project. Her effort in managing all the projects in the group and enabling the collaborative effort really helped me in achieving my goals easily. I am extremely grateful to Alet, for being a great supervisor and colleague. I have learned and improved a lot all thanks to his constant support, guidance and advice. I would also like to express my sincere gratitude to Wim who has guided me through the project and provided me with valuable advice. His support and positive criticism have helped me improve my skills as a professional.

I would like to extend my deepest gratitude to Karthik and Shobhit, both of whom were critical members and part of the CM-HVAC-FDD and TKI low ΔT syndrome project. The collaborative effort between the three of us have helped shape a wonderful product. A special thanks to Shobhit who has been a great colleague and friend, without whose support I would not have achieved my goal.

Finally, I would like to thank my family and Nivya, who have constantly supported and motivated me throughout my PDEng. They have been the most supportive of my work and career goals, and I would not be in the position I am without their love and support. I dedicate this thesis to them.

Anand Thamban

August 2022

Abstract

The low ΔT syndrome is a phenomenon which affects the performance of the chilled water system in buildings, and if left unaddressed, can have a significant negative impact on the energy consumption of the system and human comfort. The syndrome is observed with a reduced return water temperature and an increased mass flow rate through the cooling coil. Both phenomena occur when certain faults are present in the air handling unit and/or chilled water system, which lead to a reduced cooling output. As a result, the energy consumption of the pumps and chiller increase, to account for the reduced cooling output. In some instances, the system is not able to meet the cooling requirements hence leading to human discomfort. Most of these faults cannot be easily detected by the building management system and are hence not reported to the operator. The issues come to the attention of the operator only when severe comfort complaints or increased energy bills are observed. To prevent this, it is necessary to detect and diagnose the low ΔT syndrome at an early stage, so that the faults which cause it can be fixed at an early stage, leading to energy savings and better comfort.

Since there are more than 20 faults that lead to the low ΔT syndrome, the most impactful ones have been selected using a Pareto analysis. The analysis was conducted with the help of a building simulation software called EnergyPlus, and it was found that the stuck valve fault and reduced supply air temperature fault had the largest impact on the energy consumption of the system. A fault detection and diagnosis tool was developed which can detect the low ΔT syndrome and diagnose the specific fault which caused it, at an early stage. This helps in reducing the unnecessary energy consumed during the period when the fault goes unnoticed for a long period of time. The low ΔT syndrome is detected using XGBoost based regression algorithms which detect anomalies in the cooling coil valve position and return water temperature. The faults are diagnosed using a diagnostic Bayesian network, which is based on the 4S3F (4 symptoms 3 faults) framework, where the diagnostic results are provided in a probabilistic manner.

The tool was developed and validated using the data from two case study buildings in the Netherlands. It was validated during a final test conducted by introducing the stuck valve and reduced supply air temperature faults into the system. The developed tool was able to successfully detect the low ΔT syndrome with a detection accuracy of 96% and diagnose the faults by labelling them with a diagnosis accuracy of 93%.

Table of contents

Nomenclature

FDD – Fault Detection and Diagnosis
HVAC – Heating, Ventilation and Air Conditioning
AHU – Air Handling Unit
BMS – Building Management System
WP – Work package
ML – Machine Learning
AI – Artificial Intelligence
DBN – Diagnostic Bayesian Network
4S3F – 4 Symptoms 3 Faults
MFR – Mass flow rate
RWT – Return water temperature
SWT – Supply water temperature
SAT – Supply air temperature
RAT – Return air temperature
CCVP – Cooling coil valve position
CAV – Constant air volume
DOE – Department of Energy (United States of America)
EMS – Energy Management System
P-S – Primary-Secondary
APAR – Air handling unit performance assessment rules
CPT – Conditional Probability Table
CHW – Chilled water
ANN – Artificial Neural Network
SVR – Support Vector Regression
XGBoost – eXtreme Gradient Boosting
UI – User interface
UX – User experience
P&ID – Process and Information Diagram

Executive summary

Project summary

With the rising global temperatures around the world due to climate change, there is an increasing demand for cooling in indoor spaces to ensure that occupants can live and work in a comfortable environment. To ensure this, the chilled water systems in buildings are operated more frequently than usual to comply with the comfort requirements. With the prolonged and more frequent use of the equipment, the chance of faults occurring in the system increase. One such phenomenon which affects the performance of the chilled water system in buildings/distribution plants is the low ΔT syndrome. The main characteristics of the low ΔT syndrome are a reduced return water temperature and an increased mass flow rate through the cooling coil. These characteristics occur when certain faults are present in the system, which lead to a reduced cooling output. The consequences of the low ΔT syndrome are an increased energy consumption and/or inability to meet the cooling requirements leading to discomfort. To avoid both issues, a fault detection and diagnosis tool has been developed to detect the low ΔT syndrome swiftly. Bayesian networks are used to diagnose the various faults which can cause the low ΔT syndrome.

Product

The tool is developed as a larger continuous monitoring tool to detect and diagnose faults in the HVAC system, with more focus on the cooling, heating and heat recovery system. The special focus of this project is on the development of an algorithm to detect the low ΔT syndrome. The tool is designed as an online webpage which can be hosted locally or on the cloud. It consists of multiple sub-pages for main alarms, diagnostic Bayesian network analysis and machine learning analysis. The tool is intended for use by multiple types of end users including HVAC experts, facility and building managers and machine learning experts, hosting special features for each of the end users. The tool provides a simple and easy to understand alarm system for when the low ΔT syndrome has been detected (red indicator for faulty conditions and green indicator for normal conditions), with multiple to-do actions and interactable graphs to assist the user in the final decision-making process.

Introduction

_____

This chapter introduces the problem related to the low ΔT syndrome and is divided into three sections. In section 1.1, the problems related to the low ΔT syndrome are discussed along with the research related to its detection and diagnosis. Section 1.2 describes the overall methodology with the division of tasks into structured work packages, with each of the subtasks listed as sections of the different chapters in the thesis. Finally, in section 1.3, the layout and organization of the thesis is explained.

1.1 Problem description

The built environment contributes to about 37% of the total energy consumption in the Netherlands (RVO, 2021). With an increasing trend of warming witnessed every year, the cooling demand is expected to increase in the European continent (Lhotka et al., 2018). To cope with this rising cooling demand, the energy consumption for cooling is also expected to increase. Around 63% of this cooling energy use comes from Heating, Ventilation and Air-Conditioning (HVAC) systems (Marquart & Lange, 2017). The HVAC system aims to maintain thermal comfort and the required indoor air quality for human occupation. Research has shown that recommissioning of HVAC systems can lead to a 10-20% savings in energy use (Friedman & Piette, 2001; Mills, 2011).
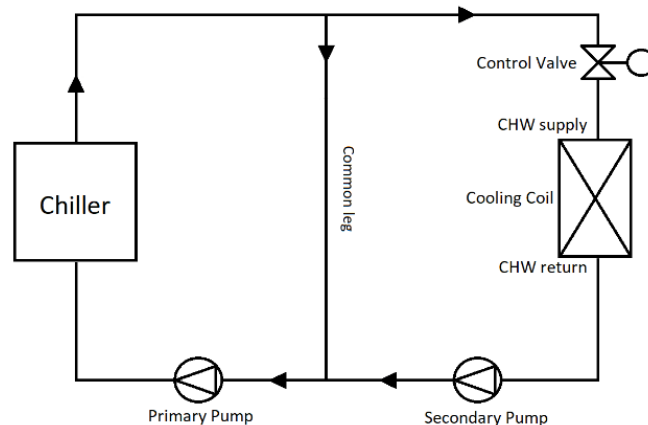
Figure 1: Primary-secondary chilled water system

In an Air Handling Unit (AHU), cooling and dehumidification of the air is achieved by a cooling coil, where the cooling capacity is delivered by chilled water (CHW) which is supplied by a chiller system, an Aquifer Thermal Energy Storage (ATES), or a combination of both as used in district cooling systems. The most common type of chiller flow system used is the primary-secondary flow system (P-S) as shown in Figure 1. The P-S system typically consists of a constant flow primary circuit and a variable flow secondary circuit. This system is an improvement from the traditional constant flow system which has a lower efficiency compared to the P-S system (Gao et al., 2016). The efficiency of the system, which is defined as the ratio of the cooling energy output to the electrical input, is mainly determined

by the cooling capacity provided by the chiller and the capability of the cooling coils to supply the required load. The performance of the cooling coil is mainly attributed to the water-side temperature difference and the mass flow rate (MFR) during part-load conditions. A smaller temperature difference between the supply water and return water temperatures (RWT) will lead to an inefficient chilled water system, reducing the cooling output and causing energy wastage to operate extra chillers and pumps to keep up with demand (Kirsner, 1996). This phenomenon of a reduced temperature difference across the cooling coil with an increased demand of MFR to keep up with system demand is called the low ΔT syndrome. The low ΔT syndrome is a problem that affects the performance of the cooling coils and eventually the chiller. Addressing this problem can be a solution to improving the efficiency of chilled water systems and improving comfort conditions.

One of the ways to identify and eliminate the low ΔT syndrome is by using Fault Detection and Diagnosis (FDD) systems. FDD systems are used in the maintenance of building installations with the main purpose of detecting faults and diagnosing them, so that corrective measures can be taken to solve the occurring faults in the system. The aim of an FDD system is to detect a fault (here, the low ΔT syndrome) by observing certain signs (e.g., reduced RWT and increased MFR) and then diagnosing the causes (e.g., stuck cooling coil control valve, reduced supply air temperature setpoint) leading to the fault. Few studies have been conducted that used data-based and knowledge-based grey box models to detect the low ΔT syndrome for e.g., using simplified cooling coil models (Yan et al., 2018) and comparison of performance indices (Gao et al., 2012, 2014, 2016). In these studies, the FDD algorithms were developed for specific faults using methods which cannot be easily generalised and scaled for commercial application. Therefore, it is seen that there are no studies that have worked on an FDD tool which can be easily scaled for different HVAC systems.

The aim of this project is to design and develop an FDD tool to detect and diagnose the low ΔT syndrome, which can then be easily integrated with building management systems (BMS) (in this project, the BMS of Kropman Installatietechniek B.V.: named InsiteView), for continuous monitoring, and can be easily scaled up for commercial application. The FDD tool is developed as a larger continuous monitoring tool to detect and diagnose faults in the HVAC system, with more focus on the cooling, heating and heat recovery system. This project is contributed by three PDEng trainees. The contributions related to the heating and heat recovery system are developed by one PDEng trainee, whereas the basic prototype of the FDD tool was developed by another. The projects are supported by Rijksdienst voor Ondernemen Nederland (RVO), TKI Urban Energy and Eindhoven Engine. The end goal of all the PDEng trainees is to develop a product design that could later be further developed for commercial expansions for the benefit of the partner company and stakeholders involved.

1.2 Design methodology and work packages

To realize the development of the FDD tool, the following design approach has been developed as shown in Figure 2, where the project is divided into six work packages (WP). As a part of the development process, three different case study buildings were used as show in Table 1. The buildings include two simulation models and two real case study buildings, where a simulation model was made of one of the real buildings. A simulation model of a 5-zone small office building and an office building in Breda, was used for the analysis of the low ΔT syndrome. The office building in Breda and a school building in Nijmegen were used for the development of the FDD algorithms and the validation of the

FDD tool. Table 1 also includes the different activities conducted for each of the case study buildings, along with a description of the HVAC system.

Table 1: Case study buildings

Case study building	5-zone CAV office building	Office building in Breda	School building in Nijmegen
Type	Simulation	Simulation and real	Real
Activities	• Fault impact analysis	• Fault characterization analysis • Fault experiments • FDD tool development • FDD tool validation	• Fault experiments • FDD tool development
AHU	One central AHU for five zones	One central AHU for three zones, with cooling coil for each zone	Two AHUs operating in parallel for the whole school
Ventilation system	CAV	CAV	CAV
CHW system	Constant primary – Variable secondary with three chillers staged in parallel	Constant primary only with one chiller.	Variable primary – variable secondary system with ATES and heat pump.

The simulation models of the 5-zone building and the office building in Breda were used for the fault impact analysis and the fault characteristic analysis respectively. The office building in Breda and school building in Nijmegen were used for conducting experiments and introducing faults in the HVAC system to reproduce the low ΔT syndrome. The fault experiments for the school building in Nijmegen were conducted by a PhD researcher and were therefore not carried out during the timeline of this project. The data from both case study buildings were used for the development of the FDD tool and the office building in Breda was used for the validation of the tool.
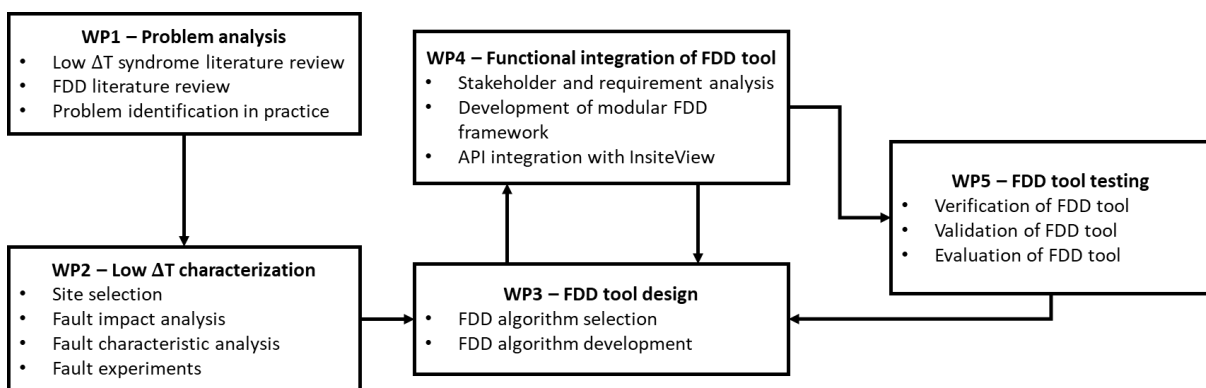

Figure 2: Project work packages and methodology

Problem analysis (WP1)
In WP1, the low ΔT problem was thoroughly analysed and studied from both scientific literature as well as data from case study buildings. This was done to get a thorough understanding of possible faults which can cause the low ΔT syndrome. The data analysis helps to understand if it is possible to

successfully identify the low ΔT syndrome without fault labels or historical fault records, with high accuracy. The different FDD methods were also studied from scientific literature to understand the possible methods which could be used to detect the fault.

Low ΔT characterization (WP2)

The characterization of the low ΔT syndrome was done as a part of WP2, to identify the most impactful faults in terms of energy consumption and comfort. It was also done to understand the characteristics of the specific faults in terms of MFR increase and RWT decrease. The most impactful faults would then be included in the FDD tool. The characterization was conducted using EnergyPlus where two case study buildings were analysed. Fault experiments were conducted where the identified faults were introduced into an operational HVAC system of a case study building to replicate the low ΔT syndrome and produce labelled faulty and fault-free data. This dataset was then used for the development of the FDD tool.

FDD tool design (WP3)

The most suitable fault detection and fault diagnosis methods were chosen from the literature review done in WP1, based on the general requirement of being scalable and generalisable. Different regression-based machine learning (ML) algorithms (Support Vector Regression, Neural Networks, Ensemble methods) were analysed and compared, and the most suitable algorithm was selected for the fault detection module. The labelled dataset from WP2 was used for the comparison and to test the ability of the model to detect the low ΔT syndrome. A diagnostic Bayesian network was used as the fault diagnosis module.

Functional integration of FDD tool (WP4)

The different modules (data pre-processing, fault detection, fault diagnosis) designed in WP3 were integrated into one tool in a modular format to ensure replaceability and scalability. The tool was designed based on the requirements of the different stakeholders of the project. Prototypes of the tool were developed in a prototyping platform called Figma, where the users' perspective into design was considered with the help of interviews by following the agile software development method. The final tool was developed in Python using a dashboarding platform known as Dash.

FDD tool testing (WP5)

The final version of the FDD tool was verified against the stakeholder requirements and validated by conducting in-situ live testing where faults were introduced into the system. The tool was evaluated for the different fault cases where its capabilities, drawbacks and limitations were analysed.

1.3 Thesis layout

Based on the different work packages and the methodology, the thesis has been laid out in the following order:

- **Chapter 2 – Problem definition: Low ΔT syndrome.** This chapter contains detailed information about the low ΔT syndrome, providing more insights into the specific causes of the phenomenon and their consequent impact on the system. A brief introduction to FDD systems is also presented. The results from this chapter include the identification of the most impactful faults, which is then further used in chapter 3 for FDD method selection.

- **Chapter 3 – Working phase: Algorithm development**. This chapter gives more information about the selection of the required FDD methods and development of the algorithms. The developed algorithms are then used as back-end components in the final product, which is explained in chapter 4.
- **Chapter 4 – Selection phase: Product features.** This chapter sets out the complete product development process with a focus on stakeholder requirements analysis, the product architecture and the front-end layout.
- **Chapter 5 – Shaping phase: Product development.** This chapter discusses the verification of the product in terms of the stakeholder requirements and includes a validation with a live in-situ testing of the FDD tool followed by an evaluation of the tool.
- **Chapter 6 – Discussion.** This chapter provides an analysis of the different modules of the FDD tool, including the limitations and drawbacks.
- **Chapter 7 – Conclusion.** This chapter concludes the project, providing recommendations for future work and other information regarding relevant publications related to this research.

A business plan is also proposed (presented in Appendix A9) with a financial analysis for the possibility of commercial deployment of the FDD tool.

Problem definition: Low ΔT syndrome & FDD methods 2

_____

In this chapter, the low ΔT syndrome and FDD methods are discussed in more detail. In section 2.1, a scientific literature summary of the low ΔT syndrome is discussed explaining the characteristics of the phenomenon and the different kinds of faults leading to it. The different FDD methods which could be used to detect the low ΔT syndrome are studied in section 2.2. In section 2.3, the characterization of the low ΔT syndrome is done including a fault impact analysis and a fault characteristic analysis. Finally, in section 2.4, the fault experiments conducted in the use case buildings are explained.

2.1 Literature summary: Low ΔT syndrome

The low ΔT syndrome is an infamous phenomenon related to the CHW system in a building/distribution plant. The problem was noticed during the 1980s when the P-S chilled water system was widely used. It was even noticed in the famous NASA Johnson Space Centre in USA, where a central CHW plant supplied CHW to 40 buildings on the campus. The issue was that even though the plant was designed for a ΔT of 8.9 °C across the central plant chillers, the system could attain only an average ΔT of 3.9 °C, with ΔT of 5.6 °C at its best (Kirsner, 1995).

A CHW system is designed to meet the demand load of the building(s) by supplying CHW with the required cooling capacity. This cooling capacity is defined by the waterside temperature difference and the MFR. In the instance when certain faults occur in the system, the RWT would be lower than usual. This leads to an increased demand of MFR to keep up with the required cooling capacity to cool the air. This smaller temperature difference between the supply water and return water will lead to an inefficient chilled water system, reducing cooling output and causing energy wastage to operate extra chillers and pumps to keep up with demand (Kirsner, 1996). This phenomenon of a reduced temperature difference across the cooling coil with an increased demand of flow to keep up with system demand is called the low ΔT syndrome (Kirsner, 1996).

The issue with the low ΔT syndrome is that MFR and the cooling load do not keep up with each other, hence requiring additional chillers to maintain the flow requirements even though they might not be fully loaded (Taylor, 2002), i.e. not running at its full capacity. Otherwise, the flow direction in the common leg would reverse (see Figure 1). The subsequent result is the increased unnecessary energy consumption of the chiller and pump, and/or the failure to meet the cooling loads, hence leading to occupant discomfort. To prevent the low ΔT syndrome from happening, it is important to understand what are the causes which lead to the degradation of ΔT. Table 2 shows a selection of possible faults which have been identified (Dai et al., 2021; Taylor, 2002). These various faults have been classified as design faults, abrupt faults or incipient faults. This is done in order to identify which faults can be detected from continuous monitoring and data analysis methods.

Table 2: Classification of known causes of the low ΔT syndrome

Design faults	Abrupt faults	Incipient faults
Use of 3-way valves	Reduced supply air temperature (SAT) setpoint	Reduced coil effectiveness through fouling
Improper coil selection	Supply air temperature sensor offset	Air filter fouling
Improperly selected control valves	Cooling coil control valve failure	
No control valve interlock	Supply fan control failure	
Improperly piped coils	High chilled water supply temperature	
Outdoor air economizers	Improper PID parameter settings	
100% outdoor air systems	Increased supply water temperature (SWT)	
Unbalanced water loops		
Oversized pumps		

Table 2 shows that most of the causes of the low ΔT syndrome are design faults and abrupt faults. Design faults on one hand can be avoided during the design or installation phase of the HVAC system and can also be avoided once commissioned but with more complexity. Abrupt faults on the other hand can be fixed during operation since they are mostly failure or control faults. Incipient faults occur over a period of time and cannot be suddenly detected. Faults like coil fouling due to biocontamination usually occur in hot and humid climates like in Florida (Firrantello et al., 2018) and is not often observed in a moderate climate like in the Netherlands. A detailed 8-year analysis of cooling coils in an office building near Schiphol showed that the effectiveness of the coil did not reduce and fouling was unlikely also due to the fact that the coils were epoxy coated to prevent corrosion from salty air (Sembian, 2019). Air filter fouling is also prevented in most of the installations using predictive maintenance techniques and timely replacement of filters. The focus of this study is therefore mostly on abrupt faults.

As a part of a continuous monitoring system, an FDD tool can mostly detect abrupt faults since it uses anomaly detection/ pattern recognition models. Design faults being inherent faults which exist from the beginning of system life are rather difficult to be detected using data-based FDD methods. A more detailed explanation of the different FDD methods is explained in the next section. A more detailed analysis of the different abrupt faults, their characteristics, and their impacts are explained in section 2.4.

2.2 Literature summary: FDD methods
FDD tools play an important role in improving building energy efficiency and reducing equipment downtime, energy penalties, and service costs (Zhao et al., 2019). Malfunctions and degradations of sensors, actuators, controllers and other HVAC components, which can lead to an uncomfortable indoor environment and/or increase the total HVAC energy consumption can be detected using FDD

tools. These FDD tools can be commissioned to monitor the data obtained from the BMS and provide insights to a user if certain faults are detected.

FDD is split into two parts, fault detection and fault diagnosis. Fault detection is the process of identifying if a fault is present in a system by observing certain symptoms or anomalies. Fault diagnosis is the process of identifying the exact fault present in the system and its location. Generally, both fault detection and fault diagnosis methods can be split into data driven-based and knowledge driven-based methods.

2.2.1 Fault detection

Figure 3 shows the classification of different fault detection methods. Knowledge driven-based methods generally rely on domain knowledge where fault detection is based on simple rules having clear physical meaning. The most common ruleset used is the air handling unit performance assessment rules (APAR) (Schein et al., 2006). They are generally easier to develop and the rule sets can be easily expanded (Bruton et al., 2014). In contrast, data driven-based models work by detecting changes in patterns or anomalies in the measured data. They usually require a sufficient amount of data to function properly and can be easily deployed to other HVAC systems as well (Bruton et al., 2014).
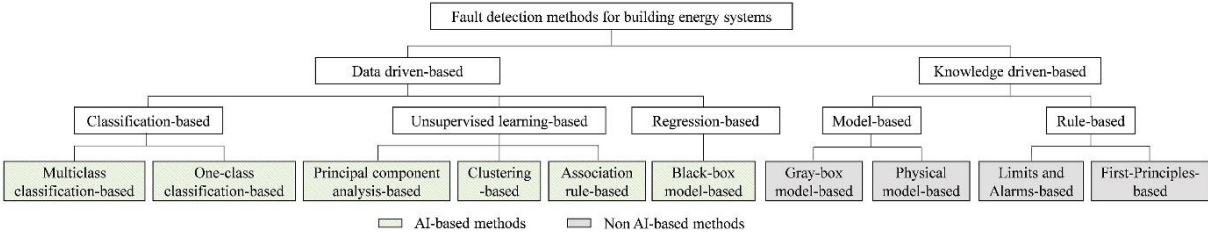

Figure 3: Classification of fault detection methods for building energy systems (Zhao et al., 2019)

Under the umbrella of data-driven methods, classification-based methods work on the principle of assigning data into specific classes, where data can be assigned as either faulty or fault-free (one-class classification) or be assigned into multiple fault classes (multi-class classification). For this approach, there is a need for a well-developed dataset with labelled faulty and fault-free data points. Since this is not practical for large scale commercial applications, classification-based methods are ruled out.

Unsupervised learning-based methods, which include different clustering-based methods, try to detect faults by splitting the data into different clusters based on their statistical characteristics (Zhao et al., 2019). The drawback with this method is that the clusters would not be labelled for faults specific to the low ΔT syndrome making it difficult for fault diagnosis.

Regression-based methods use anomaly detection as a means of detecting a fault, where the model, which is trained to predict fault-free data, detects a deviation from the measured value and signals a fault. This method is more suitable for situations where it is not possible to develop faulty training data sets and labelled data. This approach is more scalable with better chances of commercial development.

Figure 4 shows a flow diagram of how regression-based fault detection methods work. In this method, an ML model is trained and developed offline using pre-processed fault-free operational data of the building. The ML model, therefore, predicts fault-free operational data. During operation of the FDD tool, the ML model makes a prediction of the variable of concern and a comparison is made between

the predicted value and the measured value. A fault is then detected if the residual values (predicted value – measured value) are greater than a specified threshold. Once a fault is detected, the data set is sent to the fault diagnosis module.
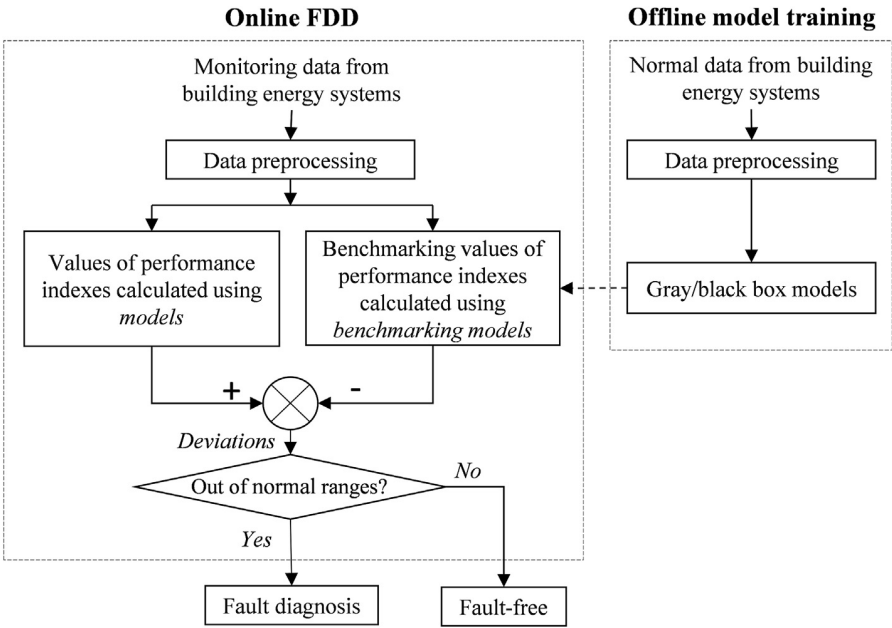

Figure 4: Flow diagram of regression-based fault detection methods (Zhao et al., 2019)

2.2.2 Fault diagnosis

The classification of fault diagnosis methods is very similar to fault detection. Figure 5 shows the classification of different fault diagnosis methods, with the main classification as data driven-based and knowledge driven-based methods. The data driven-based fault diagnosis methods which are classification and unsupervised learning-based, rely on similarity of patterns and statistical characteristics to diagnose the fault. In this case as well, large amounts of labelled faulty data are required to achieve a good quality diagnosis, something which is not always practically and commercially feasible when considering scalability. Within knowledge driven-based methods, the inference-based method is an attractive fault diagnosis method, since expert domain knowledge is more effective than data-based approaches especially in situations where diagnostic information is incomplete and uncertain (Zhao et al., 2019). A few highlights of knowledge-based methods include probabilistic reasoning and fuzzy reasoning which is observed in Bayesian networks and fuzzy logic methods.
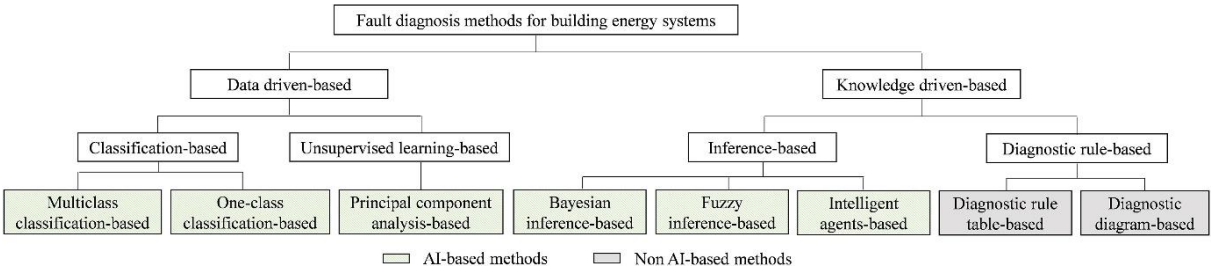

Figure 5: Classification of fault diagnosis methods for building energy systems (Zhao et al., 2019)

Fuzzy logic methods include complex "IF-THEN" rules and known facts to produce conclusions from a set of input variables. The issue with fuzzy logic method is that as the complexity of the problem grows,

the number of fuzzy rules and sets increases as well, making it difficult to adjust and tune the fuzzy sets if required.

A Bayesian network is a probabilistic graphical model that represents the relationships of probabilistic dependence within a group of variables (Zhao et al., 2019). It consists of nodes and edges, where nodes are divided into fault nodes and symptom nodes, and edges link the fault node with the symptom node. The edge which connects the nodes is directional in nature which indicates that a fault causes specific symptoms to occur, as shown in Figure 6. Each fault node has a prior probability assigned to it and the edge between the fault node and symptom node has a conditional probability assigned to it, indicating the probabilistic relationship between the fault and the symptom. The main advantage of Bayesian networks is that labelled faulty data is not required for developing the FDD tool since the probabilistic relationships between faults and symptoms are obtained from domain knowledge.
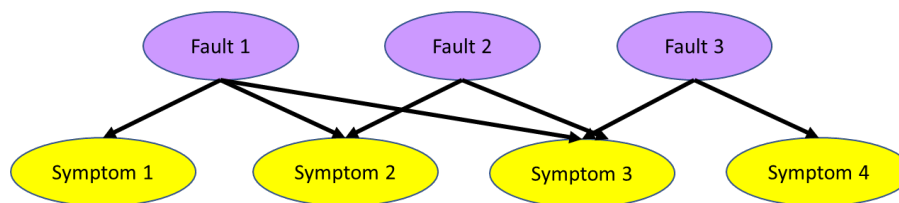

Figure 6: Schematic of a Bayesian network (Zhao et al., 2019)

2.3 Characterization of the low ΔT syndrome

The different abrupt and incipient faults mentioned in section 2.1 have varied characteristics which impact the cooling system differently. This means that each of the faults shows the low ΔT syndrome differently. During the development stage of the FDD tool, it is important to focus on those faults which have the largest impact on the system, i.e., the energy consumption of the pumps/chillers or the unmet cooling hours in the zone. The Pareto rule can be used here, where 20% of the faults account for the top 80% of the extra energy consumption in the system (Corten, 2019). The intensity of the observed low ΔT syndrome, and the subsequent impact on the energy consumption and comfort due to the different faults can be studied using building energy simulation software.

Building performance simulation software like EnergyPlus is useful to simulate operational faults in an HVAC system of a building to understand their influence on energy consumption and occupant comfort (Zhang & Hong, 2017). It has the capability to simulate different kinds of faults including fouling faults, sensor offset faults, performance degradation, control faults and stuck faults. The availability of native fault modelling objects within EnergyPlus makes it a highly desirable simulation tool compared to other commercially available software. Previous studies of fault impact analysis have already been done where air filter fouling, coil fouling and sensor offset were simulated (Zhang & Hong, 2017). This study would expand the list of faults and focus more on the low ΔT syndrome.

The characterization of the low ΔT syndrome is divided into two parts. In the first part, the simple 5-zone small office building model was used to conduct a fault impact analysis and eventually identify the most impactful faults using the Pareto rule. The 5-zone small office building is a standard validated model developed by the United States Department of Energy (DOE). Here, the most impactful faults were determined based on the influence on key performance indicators like chiller energy consumption, pump energy consumption, and unmet cooling hours.

For the second part, the identified faults were studied in more detail in a simulation model of an actual building which is also one of the case study buildings used in this project. The building simulation model of the office building in Breda was developed in EnergyPlus, with the main purpose of understanding the intensity of the faults and the corresponding increase in MFR and decrease in RWT. The simulation model was validated by comparing the chiller energy consumption, cooling coil power, MFR, and temperature difference for both high demand cooling days and low demand cooling days. Since the EnergyPlus simulation was not linked with real-time data, it was necessary to find days where the weather data in the simulation matched with real conditions. For this purpose, a comparison was made between the hourly outdoor air temperature profiles of the real data and simulation data in terms of quartiles, median and mean values, and only those specific days were chosen where the values were similar. The detailed results of the validation are shown in Appendix A1.

2.3.1 Fault impact analysis

For the fault impact analysis, the 5-zone office building was used, a 3-dimensional visualization of which is shown in Figure 7. A few changes were made to the original DOE model. For instance, a P-S chilled water system was used with three chillers staged in parallel. The building uses a CAV system, where the system provides constant airflow to the zones and the zone temperature is controlled by varying the SAT. The purpose of including these modifications was to study the low ΔT syndrome in detail, which occurs more in P-S chiller water systems with multi-staged chillers. Moreover, both the case study buildings used in this study use CAV systems. From Table 2, only the abrupt and incipient faults were considered for fault impact analysis. Even though it was determined that incipient faults like coil fouling rarely occur in a Dutch climate, it is nevertheless included in the analysis to get an understanding of its impact on energy and comfort. Out of the nine faults mentioned, only six of them were analysed since it was not possible to simulate all the abrupt and incipient fault cases in EnergyPlus. The faults were introduced using the native EnergyPlus operational faults module, the component design module, and the Energy Management System (EMS) module. The various faults and the method of introducing the fault in EnergyPlus are shown in Table 3.
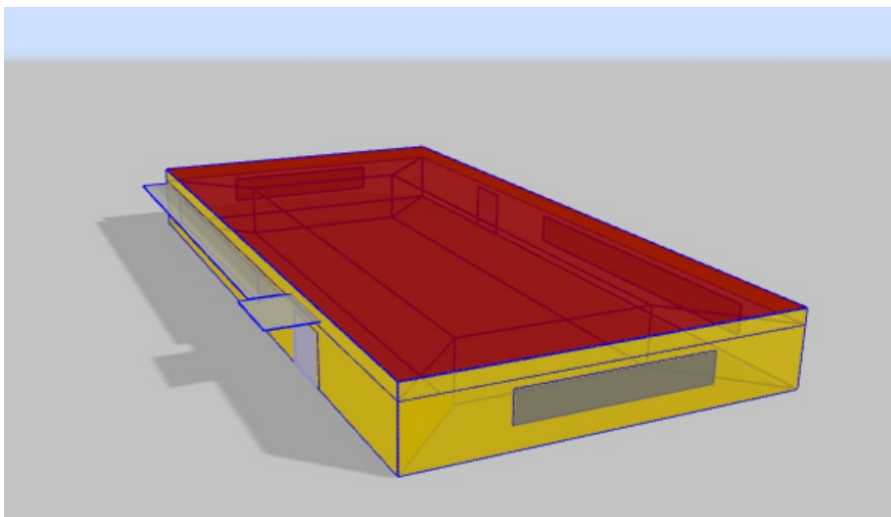

Figure 7: 3D model of 5-zone office building

Table 3: Implemented faults and methods for the fault impact analysis of the 5-zone building model

Fault type (abrupt and incipient)	Intensity	Method of implementation
Reduced SAT	Reduce by 2K.	Reduce the temperature setpoint by 2K in operational schedules.
Stuck cooing coil valve	Stuck at 75%.	Use EMS to fix the MFR actuator at 75% of max value.
Stuck fan fault	Stuck at lower (70%) and higher (130%) position compared to design value.	Set the fan flow rate to desired value in component module.
Cooling coil fouling	Reduced effectiveness by 30%.	Reduce effective UA value by 30% using operational faults module.
SAT offset	Offset of +2K.	Introduce an offset of +2K using operational faults module.
Increased SWT	Increase by 2K.	Increase SWT setpoint by 2K in operational schedules.

The main parameters influenced by the presence of the low ΔT syndrome are the secondary pump energy consumption, the chiller energy consumption, and the unmet cooling coil hours which indicate the number of hours where the comfort conditions were not met by the cooling system. The most impactful faults would be decided based on which parameters have the highest impact. The detailed results of MFR and RWT trends for each fault case are shown in Appendix A2. The comparisons of energy consumption of the pump and chiller were made for one particular day during the whole period, where the outdoor air dry-bulb temperature was between 10°C and 24°C. This specific range was chosen since it is the average daily temperature range found in the Netherlands during the cooling season (CBS, 2019). For the same day, there were no unmet cooling hours present for the baseline case, as well as faulty cases. A yearly analysis showed that unmet cooling hours occurred for less than six days a year. Therefore, an annual comparison is done for unmet cooling hours.

Figure 8, Figure 9, and Figure 10 show the percentage change in energy consumption of the chiller, energy consumption of the pump, and the unmet cooling hours respectively (annual basis). For the energy consumption, shown in Figure 8 and Figure 9, it can be seen that in almost all cases there is an increase in energy consumption, with the largest increase present for the stuck valve fault and the reduced SAT fault. The energy consumption and unmet cooling hours (shown in Figure 10) are the same for both the reduced SAT fault and the SAT offset fault. This is because in the simulation, both faults have the same net effect of reducing the SAT. Therefore, for the next set of studies only the reduced SAT fault is considered.

The energy consumption increase is the lowest for the fouling and higher airflow fault, with values less than 0.5% (except for a change in pump energy consumption due to fouling, where it is approximately 10%). Compared to the other fault cases, the increase is negligible.

For the increased SWT fault, the chiller energy consumption decreases whereas the pump energy increases. This is because the leaving chilled water temperature from the chiller is higher than normal (by 2K) hence requiring lesser energy to generate the required cooling. But on the demand side, since

the supply water temperature is higher, the pump needs to supply more water to maintain the cooling power supplied to the air, therefore increasing the energy consumption.

The energy consumption of both chiller and pump decreases when the lower airflow fault was introduced. This happens because, when the airflow rate is reduced, the cooling capacity of air reduces as well (since the SAT setpoint remains the same). Due to this, lesser water is circulated through the coil to provide lesser cooling power to the air. Consequently, the energy consumption of the pumps and chiller decrease as well.
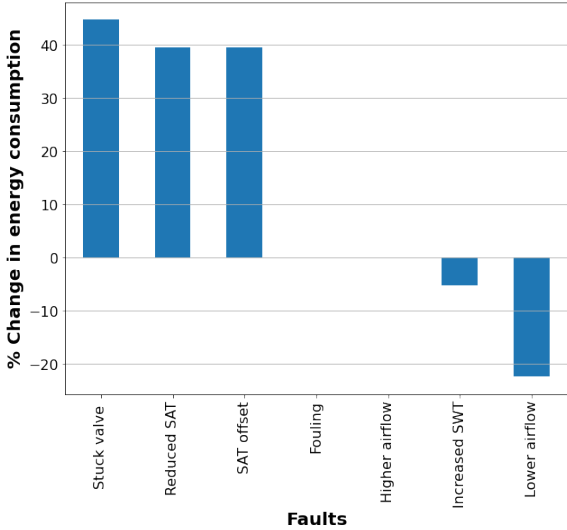
Figure 8: Percentage change in chiller energy consumption for faults compared to fault-free scenario (5338 kWh) in 5-zone building. Comparison is made for a specific day.
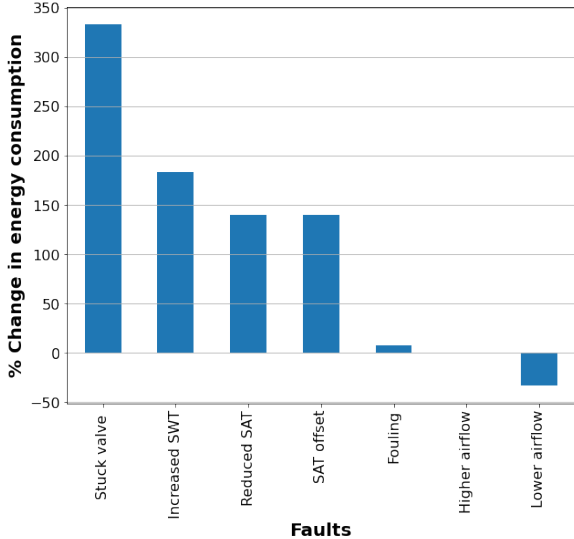
Figure 9: Percentage change in pump energy consumption for faults compared to fault-free scenario (136 kWh) in 5-zone building. Comparison is made for a specific day.

For the percentage change in unmet cooling hours (annual comparison), as seen in Figure 10, the largest increase is seen in the lower airflow fault. This is because the supply air, even though attaining the required temperature, does not have the required cooling capacity to cool the indoor air sufficiently, due to the reduced air flow rate. In this case, the supply air temperature setpoint is constant. In reality, due to comfort complaints, the SAT setpoint would be reduced to account for the reduced cooling capacity from the lower airflow fault. Only in this case, when the SAT setpoint is reduced, would the low ΔT syndrome appear in the system. It is therefore clear that the lower airflow fault in itself does not cause the low ΔT syndrome for CAV systems, since the main characteristic symptoms of increased MFR and reduced RWT were not observed.

From this study, it is concluded that among all the faults simulated, the stuck valve and reduced SAT faults are the ones ($\sim$ 20% of all faults) with the largest energy impact on the system. Even though the lower airflow fault showed signs of increased discomfort, there was no low ΔT syndrome detected in the system. Low ΔT syndrome would appear in the system only when the SAT setpoint is reduced due to insufficient cooling from the lower airflow fault.
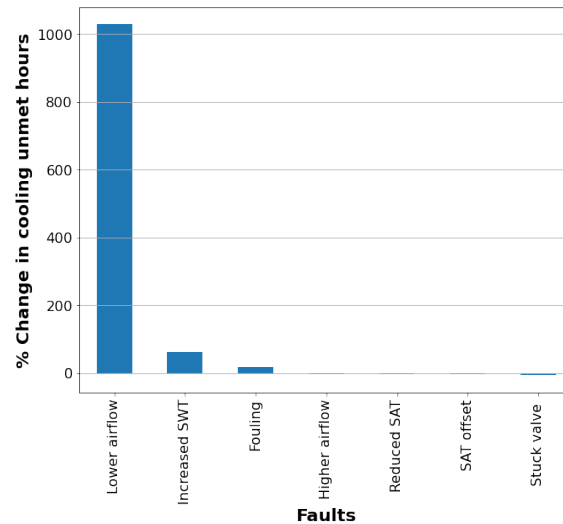
Figure 10: Percentage change in unmet cooling hours for faults compared to fault-free scenario (79 h) in 5-zone building. Comparison is made for a whole year.

The next step is to analyse the most impactful faults in detail by observing the MFR increase and RWT decrease. This helps to understand how much deviation is expected from the MFR and RWT when a fault is present. For this purpose, the fault characteristic analysis was done, as explained in the next subsection. The analysis was done for the simulation model of the office building in Breda since it would give a better understanding of the characteristics of the low ΔT syndrome when it would be simulated in the form of fault experiments at the real building.

2.3.2 Fault characteristic analysis

From the fault impact analysis, it was identified that the stuck valve fault and the reduced SAT fault are the most impactful faults in terms of energy consumption. Both these faults were introduced into the simulation model of the office building in Breda for a complete simulation period of one year. The simulation model was validated for a few days, by comparing important variables like chiller energy consumption, cooling power, waterside ΔT across the cooling coil, etc., with data from the BMS. The detailed validation is shown in Appendix A1. Figure 11 shows the 3D model of the office building in Breda, used in EnergyPlus.
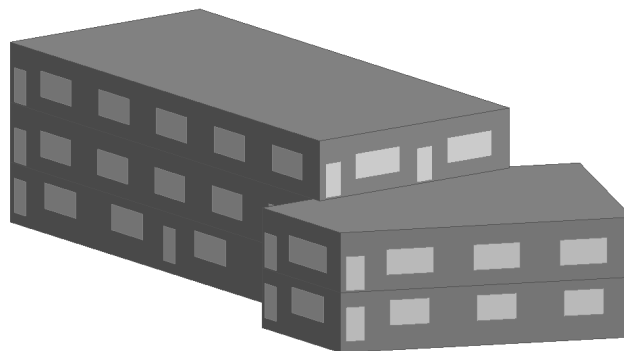
Figure 11: 3D model of office building in Breda

The modelling was done taking the properties of the thermal envelope, the division of HVAC zones, specific setpoints and schedules obtained from the BMS of the building, and the detailed design of the HVAC system into account. The model was developed in DesignBuilder and later imported to the

14

standalone EnergyPlus simulation software, to conduct the fault simulations. The focus of this study is to understand the characteristics of the low ΔT syndrome in terms of MFR increase and RWT decrease when the two identified faults are introduced.

Figure 12 shows the annual percentage change in MFR and RWT for the stuck valve and reduced SAT faults for two fault severity levels. On a yearly average, the 75 % stuck valve fault is observed to have the largest change in MFR and RWT whereas the reduced SAT – 1K fault is observed to have the lowest. Both 50% stuck valve and reduced SAT – 2K show similar trends in terms of MFR increase and RWT decrease. It is also worth noting that the MFR increases only by approximately 30% for the reduced SAT – 1K fault, whereas the other faults have values greater than 60%. Similarly, in terms of RWT decrease, the reduced SAT – 1K fault shows only an 8% change whereas the other fault show values greater than 15%. It is therefore possible that the MFR increase and RWT decrease might not be very substantial for the reduced SAT – 1K fault in an actual situation.
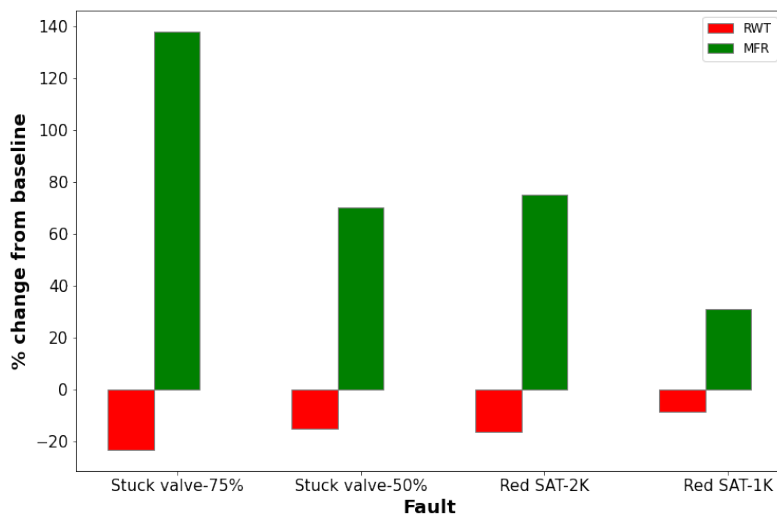
Figure 12: Annual percentage change in MFR and RWT from fault characteristic analysis

The comparison of the trends of MFR and RWT between the faulty and fault-free cases, for each of the fault cases for a specific day, is shown in Appendix A3.

2.4 Fault experiments

From the fault impact analysis, it was identified that the stuck valve fault and the reduced SAT fault have the largest impact on the energy consumption of the system. In order to successfully develop an operational FDD tool which can detect the low ΔT syndrome, it is necessary to introduce these faults into the system, reproduce the low ΔT syndrome and generate labelled faulty data so that the developed FDD tool can detect the low ΔT syndrome.

For the experiments, the office building in Breda was utilized which has also been developed as a living lab, where temperature, humidity, and MFR are measured at multiple positions in the HVAC system. The building uses a CAV system, divided into three zones: north, south and office, where each of the zones are conditioned separately by a dedicated cooling coil. The HVAC installations were retrofitted with additional sensors at multiple locations to ensure that the maximum amount of data was available for analysis. The cooling coil water circuit was also retrofitted with pressure sensors, energy meters and MFR meters on both the airside and waterside which make it ideal to study and analyse the low

ΔT syndrome in the cooling system. Figure 13 shows the simplified schematic layout of the HVAC system with a focus on only one of the three zones in the building.
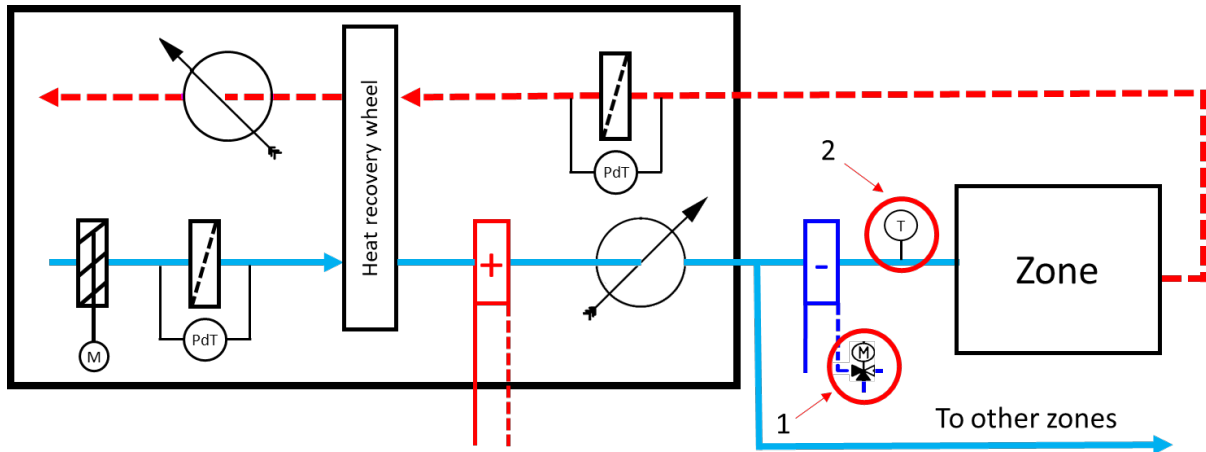

Figure 13: HVAC layout at Kropman Breda with faults indicated

Fault number 1 indicated in the figure highlights the stuck valve fault introduced in the system, whereas fault number 2 indicates the reduced SAT fault introduced in the system. For the stuck valve fault, the maximum and minimum values of the valve position were fixed to a specific value (50% and 75%) whereas, for the reduced SAT fault, the SAT setpoint was reduced by 1K and 2K. Both faults were introduced into the system for a few days during the summer of 2021. Sensor offset experiments were conducted for the same use case building by another Master's student, but these faults are not included in the scope of this study. The diagnostic performance of the developed FDD tool for sensor offset faults will be discussed in Chapter 5 in the evaluation part of the tool.

Another set of experiments was conducted at the school building in Nijmegen before the beginning of this project. Even though it was not conducted during the course of the project, it is important to mention the fault cases which were introduced in this building since the FDD tool would be developed using data from that building as well. Only the reduced SAT fault was introduced in the HVAC system where the setpoint was reduced by 2K.

The data collected during the days where fault experiments were conducted were classified as "*faulty days*", whereas the days where the system operated normally were classified as "*fault-free days*". This was confirmed by conducting thorough pre-processing of the data to remove any outliers or noise. The fault-free dataset was used for developing the FDD tool (model training), whereas the faulty dataset was used to test and check whether the FDD tool is able to detect the faults in the system.

In the next chapter, the whole process of FDD algorithm selection and development is discussed where the process of fault detection and diagnosis is explained in detail. The labelled faulty data from the experiments were used to validate the FDD algorithms.

Working phase: Algorithm development

<div style="text-align:right"># 3</div>

_____

This chapter provides a detailed explanation of the FDD algorithms used in the final FDD tool. Section 3.1 discusses the selection of the FDD algorithms based on the final product requirement. Section 3.2 provides an analysis of the chosen fault detection algorithm whereas section 3.3 provides an analysis of the fault diagnosis algorithm. The analysis of the chosen FDD algorithms uses the labelled faulty and fault-free data obtained from the results of Chapter 3.

3.1 FDD method selection

From the previous chapter in section 2.2, it was identified that regression-based fault detection is an ideal way to detect faults in the system when labelled faulty data isn't available. It was also identified that knowledge-based methods like Bayesian networks are suitable for fault diagnosis since they don't require labelled faulty data and provide better insights to the user in case of incomplete or uncertain information.

3.1.1 Fault detection algorithms

Anomaly detection is a popular way of detecting faults in the data by identifying unexpected or abnormal data from normal fault-free data. This is usually done using supervised learning regression models that predict fault-free data and is then compared with the measured data. When large residuals are identified between the expected value (fault-free) and the measured value (faulty), it can be assumed that a fault exists in the system. Different kinds of ML algorithms have been used for regression purposes including Artificial Neural Networks (ANN), Support Vector Machines (SVM) or Support Vector Regression (SVR), Decision Trees, Random Forest, and eXtreme Gradient Boosting (XGBoost). Each of these algorithms is advantageous over the other depending on the characteristics of the dataset.

SVM or SVR is an algorithm used for both classification as well as regression-based problems. It has the advantage of performing well with a limited amount of data compared to other models. But the computational time required for model development is considerably higher than other ML algorithms like ANN and Random Forest (Walker et al., 2020). Decision trees are also regression algorithms that are based on the approach of splitting a dataset while evaluating certain conditions. Ensemble algorithms are based on the ML theory that a group of weak learners create a much stronger ensemble than a single strong learner (Zhou & Liu, 2021). XGBoost is one such ensemble algorithm that has proven to be a well-performing ML algorithm in several studies (Mo et al., 2019; Pan, 2018; Yao et al., 2019) and has been previously used for fault detection in HVAC systems (Chakraborty & Elzarka, 2019). Since previous research clearly showed the benefits of ensemble algorithms compared to individual Decision Trees (Zhou & Liu, 2021), Decision Trees are not included in this study. ANN has also been used to develop regression models to predict continuous variables like energy consumption (Walker et al., 2020), temperature (Montazeri & Kargar, 2020) and cooling coil valve position (Wang & Jiang, 2004). But ANN is more complex in nature compared to SVR and XGBoost, and requires precise

adjustment of its many hyper-parameters (Seyedzadeh et al., 2018). Neural networks also perform better with larger amounts of data, which could be a drawback if limited data is available (Seyedzadeh et al., 2018).

Since each of the algorithms has its advantages and disadvantages, it is necessary to compare their performance to see which algorithm can make predictions with the least amount of error, which is an essential factor for anomaly detection. The performance of the different algorithms were compared based on performance metrics like the root mean square error (RMSE) and the coefficient of determination, which is the R^2 score. The algorithm with the highest R^2 score and the lowest RMSE would be the most successful algorithm which can detect the low ΔT syndrome.

The low ΔT syndrome can be detected using two symptoms – a decrease in RWT from the cooling coil and an increase in MFR through the cooling coil, as discussed in Chapter 2. Since MFR meters are not generally available in most of the installations, the cooling coil valve position (CCVP) was used in this study which indicates the demand for mass flow from the system. This is true for smaller HVAC systems where the pressure drop across the valve is almost always constant and the relationship between valve position and MFR is maintained. Therefore, to detect the low ΔT syndrome, two ML models need to be developed, one to monitor the CCVP and the other to monitor the RWT.

The model comparison analysis was conducted using the HVAC installation in the office building in Breda. The data required for developing the CCVP prediction model was available for two years (2020-2021) whereas the data for developing the RWT prediction model was available only for six months (April 2021-September 2021). This was because additional sensors around the cooling coil were installed during March 2021.

Figure 14 shows the methodology followed to compare the performance of the different regression algorithms. The raw data from a historical database was first pre-processed to remove noise, outliers and missing data fields. Since the low ΔT syndrome occurs only when the AHU operates in cooling mode, it is necessary to filter out only the cooling mode data. This was done by observing the cooling coil valve position ($U > 0\%$) and the chiller leaving water temperature ($T < 10 °C$).
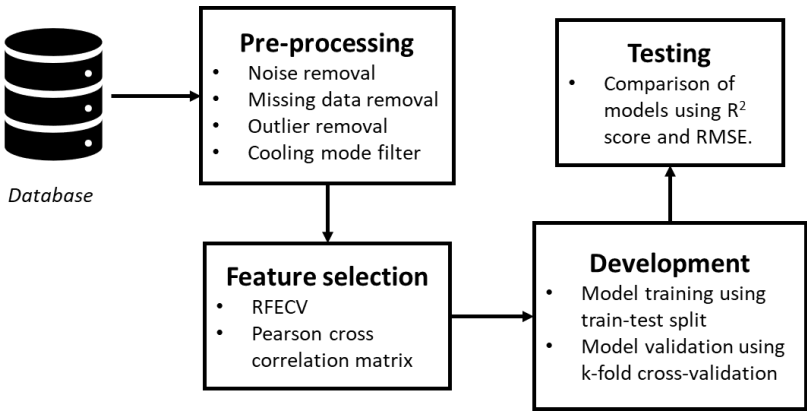
Figure 14: Methodology of regression model comparison

The processed data was evaluated to identify the required features for model development. Feature selection was done using recursive feature elimination using cross-validation (RFECV) for both XGBoost and SVR, and analysis of the cross-correlation matrix for ANN, since RFECV is not compatible with

ANNs. The algorithms were trained using a simple train-test split of 80% training data and 20% test data for the complete dataset for each prediction model (RWT & CCVP). All the ML algorithms were validated using the k-fold cross-validation method with a k value of 10. K-fold cross-validation is a commonly used statistical method to determine the performance and accuracy of an ML model. In this method, the dataset was split into k folds (groups), where one group was held as a test set and the rest (k-1) groups were held as the training set.

The model was then fit with the training set, evaluated with the test set and discarded while retaining the values. The process was then continued for the other k-1 test groups, eventually giving averaged R^2 scores and RMSE values of the model. The performance metrics of all the algorithms were then compared to choose the best performing algorithm. Certain benchmark thresholds were chosen for R^2 and RMSE for each prediction model. An R^2 score above 0.9 is desired for the predictions whereas an RMSE lower than 0.5K for RWT prediction and 5% for CCVP prediction is required. The RMSE benchmark values were identified from fault characteristic analysis for the office building in Breda where it was observed that the RWT dropped by 1K, whereas the mass flow rate increased by 0.1 kg/s ($\approx$10% CCVP). This is clearly seen in the trends of RWT decrease and MFR increase of the reduced SAT by 1K fault implemented in the simulation of the office building in Breda as shown in Figure 94 and Figure 95 in Appendix A3.

Figure 15 and Figure 17 show the comparison of the R^2 score and RMSE for the RWT prediction algorithms respectively. It is observed that the XGBoost and SVR algorithms perform well within the performance metrics well within the benchmark, whereas the scores of ANN lie just outside the threshold.
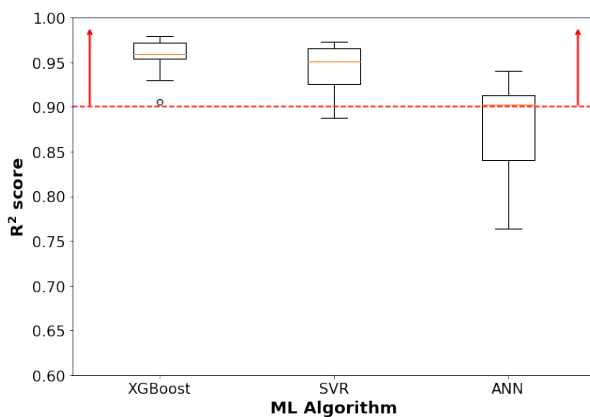
Figure 15: Comparison of R^2 score for RWT prediction
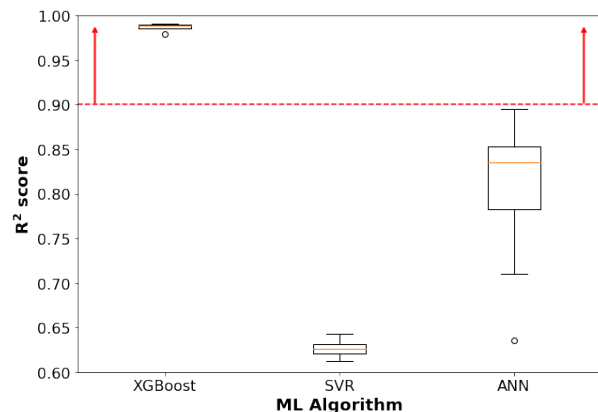
Figure 16: Comparison of R^2 score for CCVP prediction

Figure 16 and Figure 18 show the comparison of the R^2 score and RMSE for the CCVP prediction algorithms respectively. In this scenario, XGBoost is the only algorithm which performs well with the performance metrics lying within the benchmark. SVR performs quite poorly for the CCVP prediction, and this is because the dataset is much larger compared to the dataset for the RWT prediction models. The performance of ANN is relatively the same where the performance metrics are right outside the threshold value. It is therefore clear that XGBoost is the best-suited regression algorithm to be used for fault detection of the low ΔT syndrome in this study.
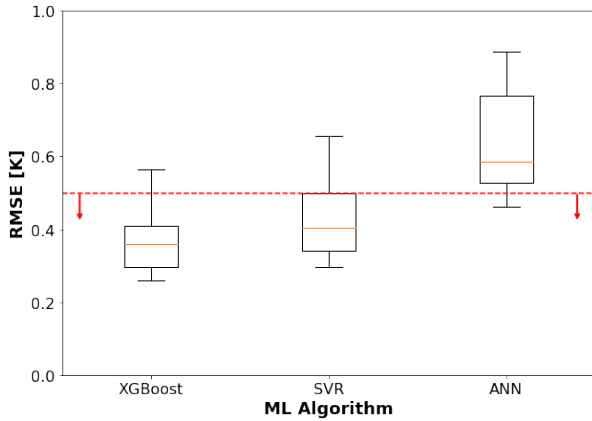
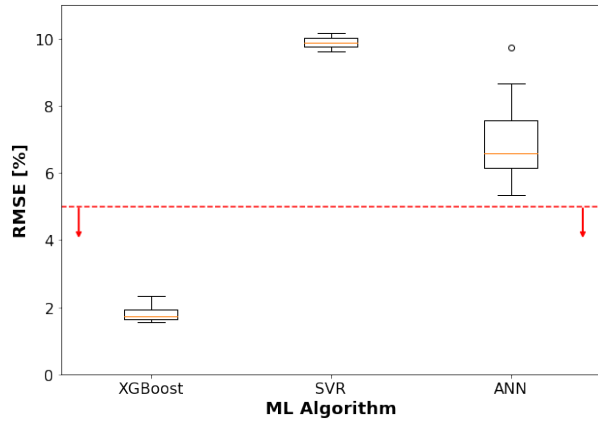Figure 17: Comparison of RMSE for RWT prediction

Figure 18: Comparison of RMSE for CCVP prediction

But for larger buildings with more complex HVAC systems, the cooling demand from one subsystem can affect the flow parameters in other parts of the system. Due to this, the pressure drop across the cooling coil and valve is not always constant, and the valve opening fluctuates to cope with the change or control strategy of the pump. Due to this, the valve position and MFR do not have the same relationship anymore. For this reason, monitoring the CCVP might not always be the best approach to identifying the demand/trends of MFR for a universal case. In this case, a virtual MFR is beneficial, especially since installing new MFR meters is not always financially feasible.

Figure 19 shows a schematic of a cooling coil with multiple sensors located around it. With temperature sensors located at the inlet and outlet of both air and water circuits, humidity sensors located at the inlet and outlet of the airside and a flow rate sensor at the airside, it is possible to calculate the MFR at the waterside using a simple energy balance as shown in Equation (1). While following this approach, it is important to understand the different modes of cooling which occur in the coil, so that accurate estimations of MFR can be made. The air entering the cooling coil can either be dry-cooled (no change in humidity ratio) or wet-cooled (decreased humidity ratio). Wet cooling occurs when the surface temperature of the cooling coil tubes is lower than the dew-point temperature of the inlet air passing through it. Dry cooling occurs when there is no change in humidity between the inlet and outlet conditions and the heat transfer is sensible.

The airside heat capacity is calculated using enthalpy h since it accounts for both sensible and latent heat transfer which occurs during wet cooling mode. The calculation of airside heat transfer using enthalpy is also done for dry cooling. The enthalpy value of air at the outlet of the cooling coil is calculated using the setpoint SAT and not the measured SAT since the setpoint defines the required value by the system and is not influenced by faults in the system. The value of RWT used in the equation is also obtained from the RWT prediction model since it can also be influenced by faults in the system. The obtained value of MFR is thus required to be fault-free just like the prediction of CCVP presented earlier.
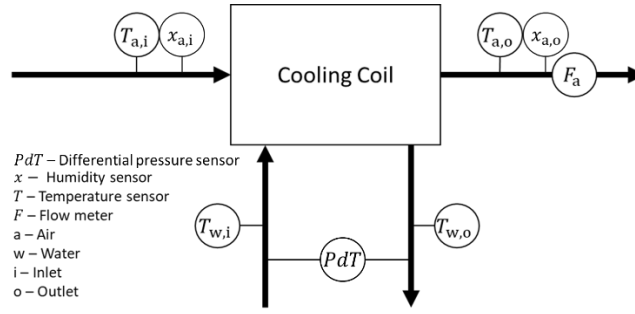
Figure 19: Required sensors at cooling coil installation for virtual MFR meter

$$\dot{m}_{\text{water}} = \frac{\dot{m}_{air}\left(h_{air,\,inlet} - h_{air,\,outlet}\right)}{c_{p_{\text{water}}}\left(T_{\text{water, return}} - T_{\text{water, supply}}\right)} \tag{1}$$

To check the fidelity of the proposed method, the virtual MFR was calculated for the office building in Breda using the XGBoost prediction of RWT. Figure 20 shows the comparison between the predicted MFR of water calculated using Equation (1), and the actual MFR measured using a meter present in the waterside circuit for the north cooling coil. The mean biased error for the virtual MFR was calculated to be 23 l/h whereas the RMSE was calculated to be 108 l/h. It is observed that the data points lie within the 25% error range for MFR values ranging between 250 l/h to 1250 l/h, and within the 10% error range for MFR values between 1250 l/h to 3500 l/h. But since the data points mostly occur within the part-load range, i.e., between 0 to 1500 l/h, it should be expected that there would be a maximum error of approximately 25% for predicting fault-free MFR values. Since this approach is a low-cost alternative to installing physical mass flow rate sensors, there exists a trade-off in terms of accuracy v/s initial installation cost. Based on the RMSE value of 108 l/h, the threshold for fault detection is set at 150 l/h, where a fault would be detected if the residual crosses this specific threshold.
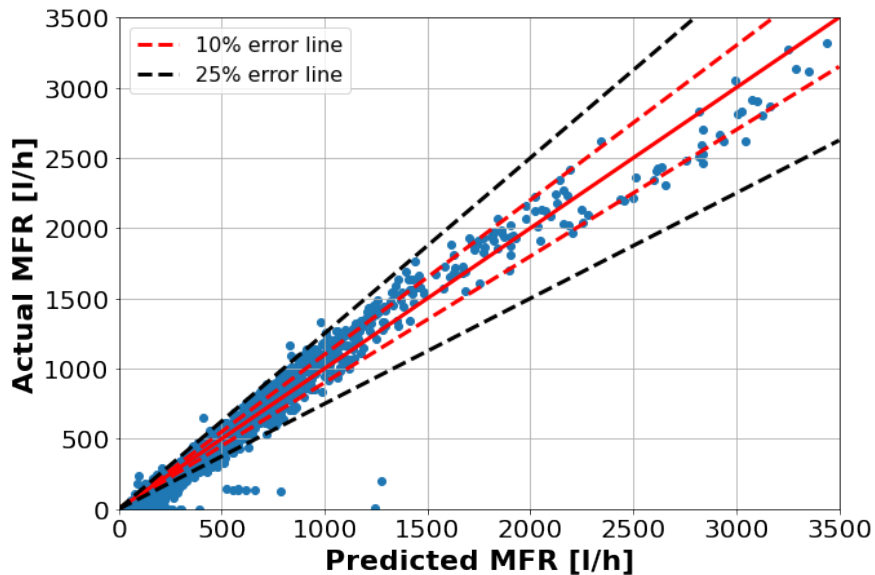
Figure 20: Comparison of predicted and actual MFR for the cooling coil of the north zone

To observe an increase in the MFR, the calculated fault-free MFR needs to be compared with a measured MFR value. For this purpose, another MFR meter needs to be developed. Differential

pressure sensors across the coil at the waterside are proportional to the MFR across the coil (Song & Ph, 2011). These sensors are easier to install in an existing installation and with the use of a balancing valve, the MFR can be measured at various cooling coil valve positions thus generating a polynomial relationship between differential pressure and MFR. Figure 21 shows a 6th order polynomial regression curve generated for MFR vs pressure difference. The regression model gives values of MFR with an RMSE of 54.4 l/h. Therefore, with data of differential pressure, it is possible to virtually calculate the MFR of water. It is also observed that for zero MFR, the pressure difference is $\sim$ 0.1-0.2 kPa. This is the static pressure difference which is observed due to the difference in height between inlet and outlet ports of the cooling coil ($\Delta P = \rho g h$, where h is the height difference, g is the acceleration due to gravity and ρ is the density of water).
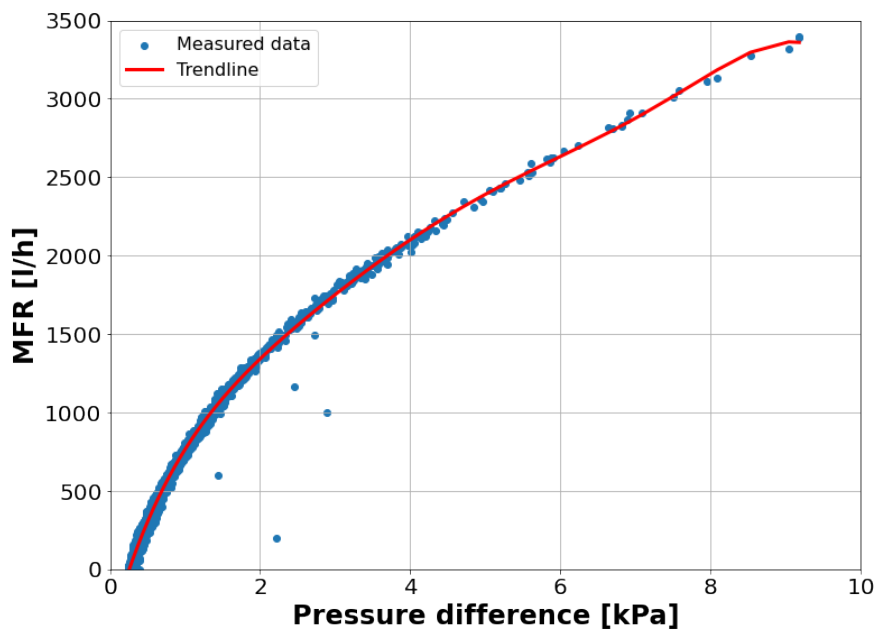
Figure 21: Polynomial regression trend line for MFR vs pressure difference for the north cooling coil

3.1.2 Fault diagnosis algorithm

From chapter 2 it was identified that a Bayesian network is a suitable knowledge-based method which provides the user with probabilistic information even with incomplete or missing information. For this study, a Diagnostic Bayesian Network (DBN) is developed based on the 4S3F method (Taal & Itard, 2020). The 4S3F method stands for 4 symptoms 3 faults, where the DBN is composed of 4 different kind of symptom nodes (balance, energy performance, operational state, additional information) and 3 different kinds of fault nodes (model, component and control). Figure 22 shows the structure of the 4S3F layout. The fault nodes in purple influence the symptom nodes in yellow with the directional arrow shown.

Figure 23 shows the architecture of the 4S3F method, where the different data flows are shown. The pre-processed raw data from the BMS is fed into different fault detection models which detect if a fault is present in the system based on user-defined thresholds. Each symptom node has a defined set of states (positive, negative, fault-free) which are linked to the different state of a fault node (faulty, fault-free) through a conditional probability table. The outcome of the fault detection models are therefore probability values (0 or 1) for all the states of a symptom node.
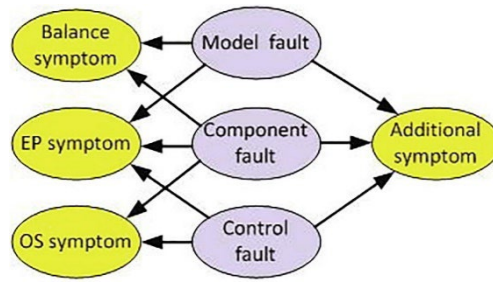
Figure 22: Structure of 4S3F DBN (Taal & Itard, 2020)
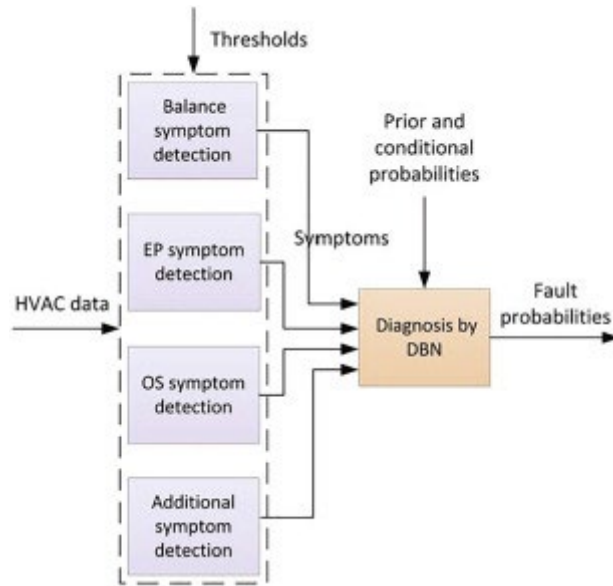
Figure 23: Architecture of the 4S3F DBN (Taal & Itard, 2020)

The algorithm of a DBN is based on the Bayes theorem of probability with its fundamental formula as shown in Equation (2). The posterior probability $P(A|B)$, which signifies the probability of a faulty A occurring in the presence of symptom B, is calculated using Bayes theorem as shown in Equation (2), where $P(A)$ is the prior probability for the fault node, $P(B|A)$ is the conditional probability between the symptom node and the fault node signifying the probability a symptom B occurs given the presence of fault A, and the probability of the symptom occurring is $P(B)$, which is obtained from the fault detection models. The different probability terms used in Equation (2) are indicated in Figure 24. The conditional probability values $P(B|A)$ are in the form of a large conditional probability table (CPT). Since the CPT for a fault node with multiple (>3) symptom nodes is quite large, the noisy-MAX approximation is used, more of which is discussed in Appendix A4.

$$P(A|B) = \frac{P(B|A) \times P(A)}{P(B)} \tag{2}$$

Figure 24 shows the DBN developed for cooling mode operation. The network includes the most impactful faults leading to the low ΔT syndrome including the stuck valve fault and the reduced SAT fault, and the lower airflow fault which can lead to comfort issues. The two ML models developed in subsection 3.1.1 are specific to detect the presence of the low ΔT syndrome, but more fault detection models are required to detect other symptoms in the HVAC system to properly diagnosis the root

cause of the low ΔT syndrome. In this context, the other models can also be termed as symptom detection models with the process being termed as symptom detection, whereas the ML models specific to detecting the low ΔT syndrome could be termed as fault detection models and the process being terms as fault detection.
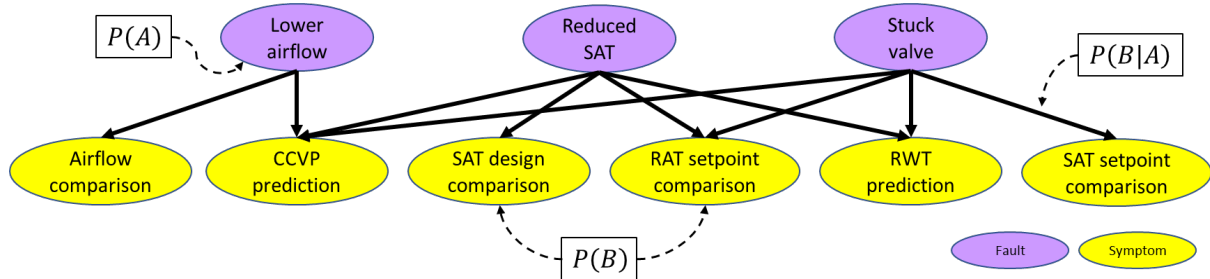
Figure 24: DBN for cooling mode

Therefore, rules-based and simple statistical models are also used to detect symptoms and provide more information to the DBN. The rules are based on the APAR set (Schein et al., 2006), obtained from literature. The different algorithms used in each of the symptom nodes are explained in Table 4. It is relevant to note that for the current DBN, only operational state symptom nodes are used since for the specific faults identified, energy balance or energy performance symptom nodes are not required. Energy balance symptom nodes would be used for sensor offset faults, which is beyond the scope of this study.

Table 4: List of fault detection models used in the DBN symptom nodes

Node name	Model type	Equation
Airflow comparison	Statistical model (Exponential weighted moving average)	$V_{measured} - V_{ewma} > 0.5 \, m/s$
CCVP prediction	ML model (XGBoost)	$U_{measured} - U_{predicted} > 5\%$
SAT design comparison	Rules based model	$T_{measured} - T_{design} > 1K$
RAT setpoint comparison	Rules based model	$T_{measured} - T_{setpoint} > 1K$
RWT prediction	ML model (XGBoost)	$T_{measured} - T_{predicted} > 0.5K$
SAT setpoint comparison	Rules based model	$T_{measured} - T_{setpoint} > 1K$

With the fault diagnosis method developed, the different fault use cases were tested for the selected fault detection and fault diagnosis method, more of which is explained in the next section.

3.2 Fault detection

The low ΔT syndrome can be detected in two ways:

- Monitoring the RWT and CCVP
- Monitoring the RWT and virtual MFR

The use of CCVP is meaningful when the pressure drop across the valve is constant, whereas a virtual MFR meter would be used when the pressure drop is fluctuating. Results from both methods are discussed below.

Fault experiments were conducted at the office building in Breda, where the stuck valve faults were introduced at two severity levels (50% stuck and 75% stuck) and the SAT was reduced as well (by 1K and 2K). The severity levels for the stuck valve fault were specifically chosen since the cooling coil valve is mostly open in the range between 25% and 56% (2nd and 3rd quartiles respectively based on a normal distribution for a large dataset), where the chance of a 50% stuck fault is much higher. The 75% stuck valve fault is considered to be a more severe fault which would happen rarely but could even happen due to control signal failure. For the reduced SAT faults, the severity levels of 1K and 2K were chosen, since it is the most probable values by which the setpoint would be reduced if comfort complaints were observed.

3.2.1 Using return water temperature and cooling coil valve position

Figure 25 to Figure 28 show the RWT and CCVP prediction for the stuck valve fault at two severity levels. The residuals for both CCVP and RWT are greater than the thresholds of 5% and 0.5K for both fault cases. The low ΔT syndrome was therefore detected for the stuck valve fault.
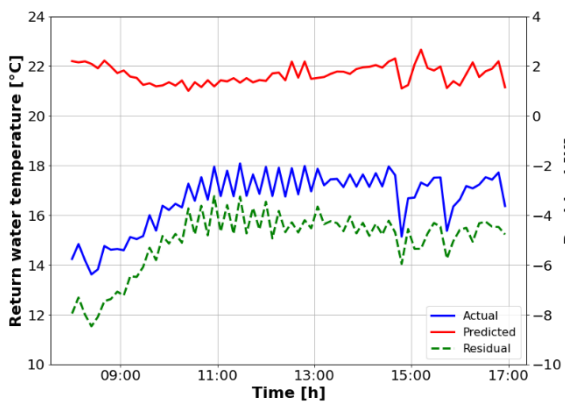
Figure 25: RWT prediction for 75% stuck valve for north cooling coil in office building Breda
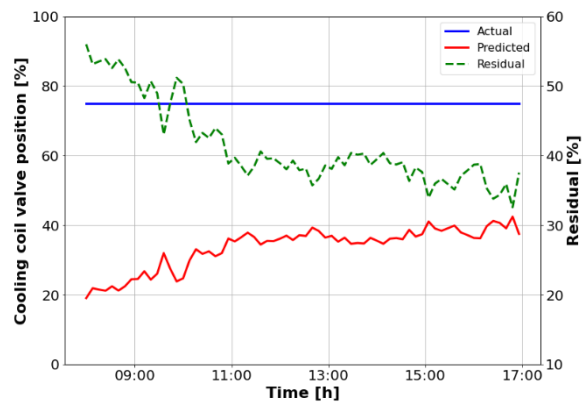
Figure 26: CCVP prediction for 75% stuck valve for north cooling coil in office building Breda
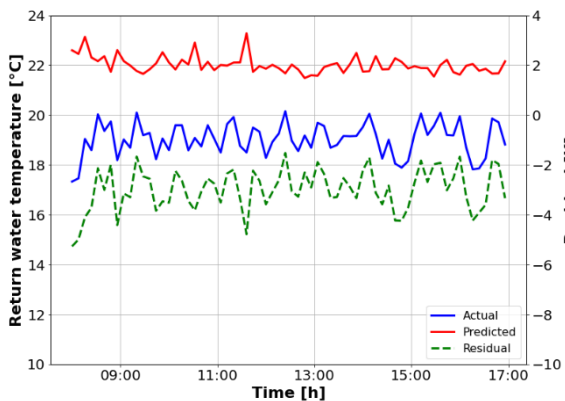
Figure 27: RWT prediction for 50% stuck valve for north cooling coil in office building Breda
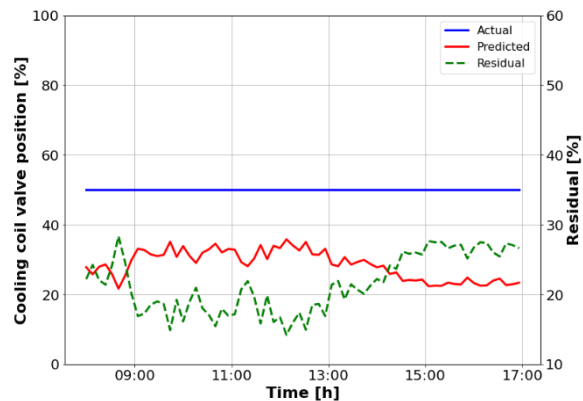
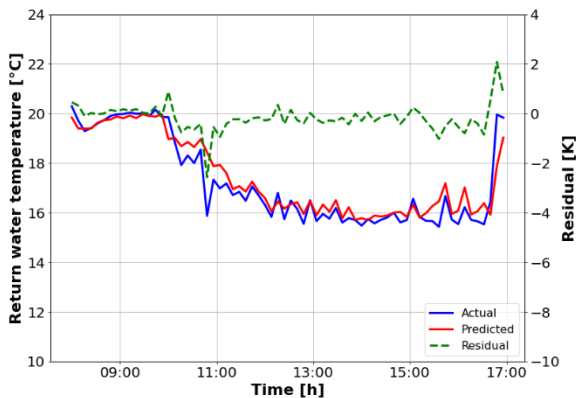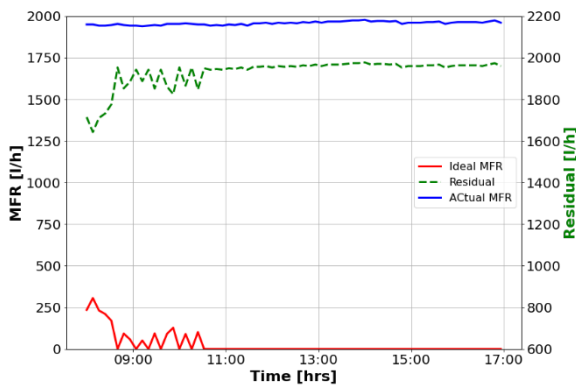Figure 28: CCVP prediction for 50% stuck valve for north cooling coil in office building Breda

Figure 29 to Figure 32 show the RWT and CCVP predictions for the reduced SAT fault for different severity levels. It is observed that the residuals for the RWT prediction is greater than the threshold

for the reduced SAT 2K fault and the CCVP prediction is greater than the threshold for the first half of the day. The low ΔT syndrome is therefore detected for the first half of the day for the reduced SAT 2K fault.

A closer look at the RWT and CCVP predictions for the reduced SAT by 1K fault show that the residuals do not cross the threshold at all. This is possible because there isn't a sufficient increase in MFR due to a reduction of the SAT by 1K. The outdoor air temperature for the days the reduced SAT faults were introduced, did not go higher than 24 °C. This could be a possible explanation as to why the predictions do not deviate much from the actual values. But based on the results from the fault characteristic analysis done in subsection 2.3.2, it was expected that the increase in MFR and decrease in RWT would not be as substantial compared to the other more severe fault cases.
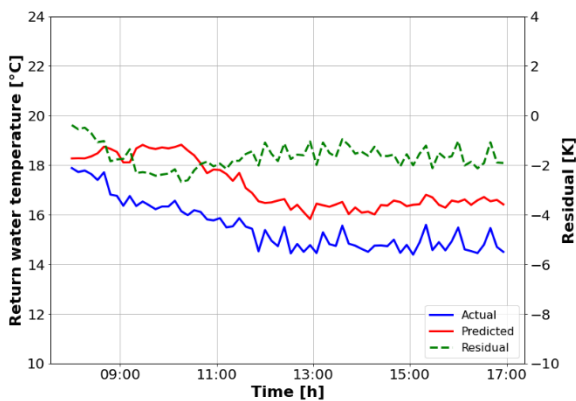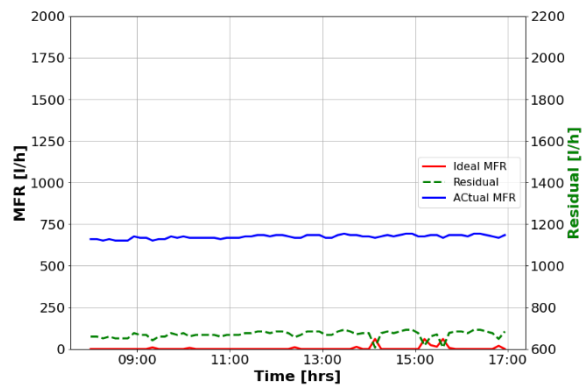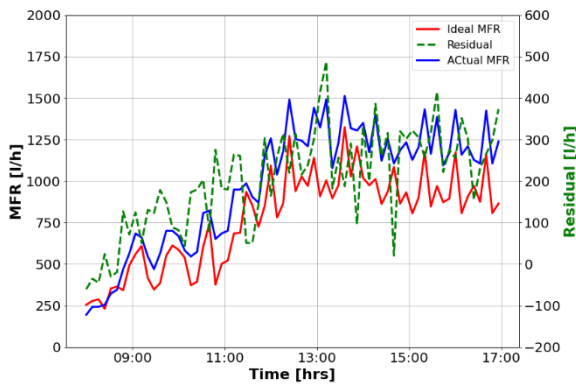

Figure 29: RWT prediction for reduced SAT by 2K for north cooling coil in office building Breda
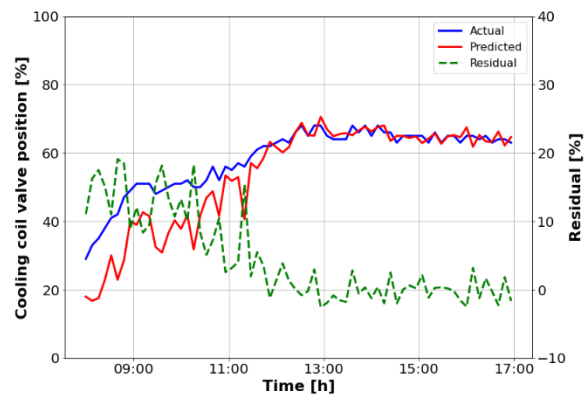
Figure 30: CCVP prediction for reduced SAT by 2K for north cooling coil in office building Breda

Figure 31: RWT prediction for reduced SAT by 1K for north cooling coil in office building Breda
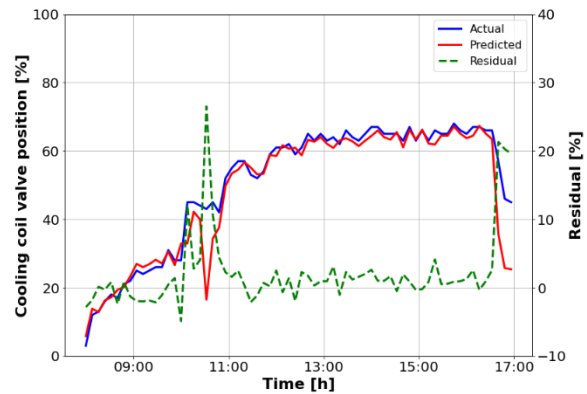
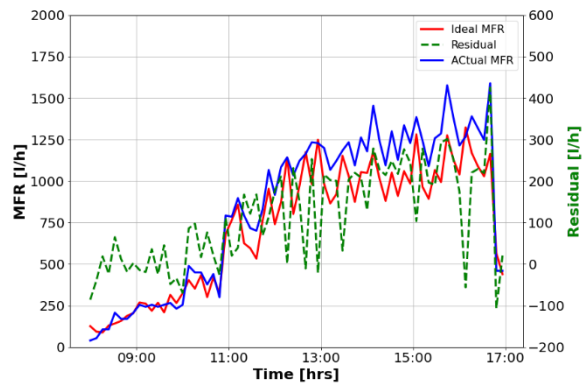Figure 32: CCVP prediction for reduced SAT by 1K for north cooling coil in office building Breda

3.2.2 Using virtual mass flow rate

The low ΔT syndrome can also be detected using the RWT prediction and calculation of MFR. Figure 33 to Figure 36 show the MFR predictions for the stuck valve and reduced SAT faults. In the case of the stuck valve faults as seen in Figure 33 and Figure 34, the ideal MFR is almost zero. This is because in the calculation of MFR, the SAT setpoint is used in Equation (1) rather than the measured SAT. Since the building is excessively cooled due to the stuck valve fault, there is no demand for cooling. Therefore, the ideal MFR in this case would be close to zero.

For the reduced SAT fault, the residuals are greater than the threshold of 150 l/h during the second half of the day for the reduced SAT by 2K fault, as seen in Figure 35. The low ΔT syndrome is therefore detected during the second half of the day, which is contrary to the results from the CCVP predictions where low ΔT syndrome was detected during the first half of the day. It is important to note that the CCVP prediction model is black-box, therefore, it is difficult to make a proper inference as to why it behaves in such a manner.

For the reduced SAT by 1K fault, however, the residuals keep fluctuating and cross the threshold intermittently during the latter half of the day. This is clearly seen in Figure 36. Therefore, the low ΔT syndrome cannot be detected for consecutive timestamps, since the RWT prediction doesn't show any sufficiently large residuals either, as seen in Figure 31.

Figure 33: MFR prediction for 75% stuck valve for north cooling coil in office building Breda

Figure 34: MFR prediction for 50% stuck valve for north cooling coil in office building Breda

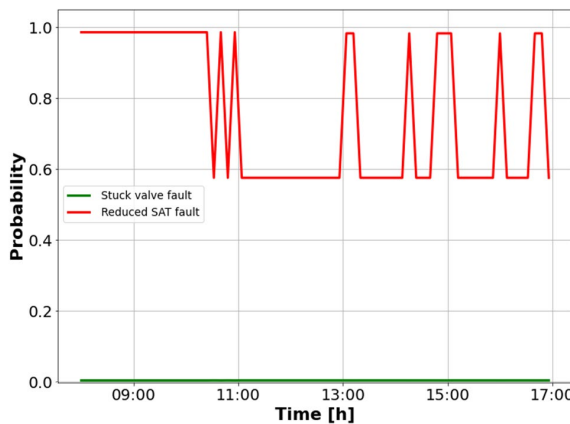Figure 35: MFR prediction for reduced SAT by 2K for north cooling coil in office building Breda

Figure 36: MFR prediction for reduced SAT by 1K for north cooling coil in office building Breda

3.3 Fault diagnosis

The output data from the different fault (symptom) detection models, including the rules based and statistical models were then inputted into the DBN, where all the states present in each of the symptom nodes were assigned a probability value. For e.g., for the CCVP prediction for the reduced SAT by 2K fault as shown in Figure 30, the different states for the symptom include positive stuck, negative stuck and fault free. Between 8 am and 12 pm, the state would be positive stuck, therefore the positive stuck state would be assigned a value of 1 whereas the other states would be 0. Between 12 pm and 5 pm, the state is fault free, therefore the fault free state would be assigned a value of 1

whereas the other states would be 0. Similarly, this process is done for all the other symptom nodes and the posterior probabilities for the different faults are calculated.

Figure 37 to Figure 40 show the posterior probability values for different fault cases introduced at the office building in Breda. The stuck valve faults are easily diagnosed with the posterior probability almost equal to 1 for both severity levels, as shown in Figure 37 and Figure 38. The posterior probability for the reduced SAT fault for the same days are 0, which means that it has not been detected in the system. For the reduced SAT by 2K fault, the posterior probabilities are larger than 0.5 and intermittently attain a value close to 1, as seen in Figure 39. During the latter half of the day, the values are closer to 0.6 since the CCVP prediction does not detect any fault during this period and therefore indicates a fault-free condition. The posterior probability values for the reduced SAT by 1K fault, as shown in Figure 40, fluctuates between 0.05 and 0.9. Even though the low ΔT syndrome was not detected, the reduced SAT fault was diagnosed in an intermittent manner due to the other symptom detection nodes present in the DBN (which are not related to the detection of the low ΔT syndrome).
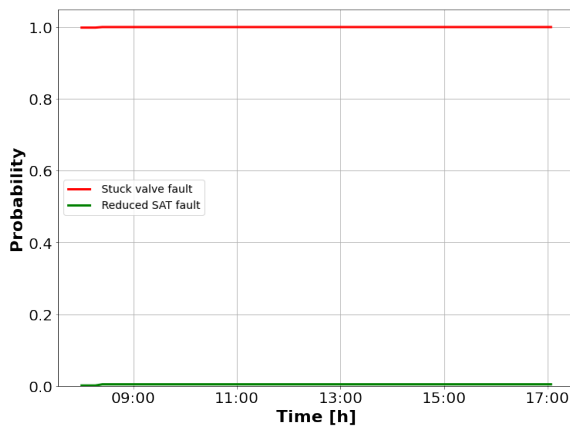
Figure 37: Posterior probability for 75% stuck valve for north cooling coil in office building Breda
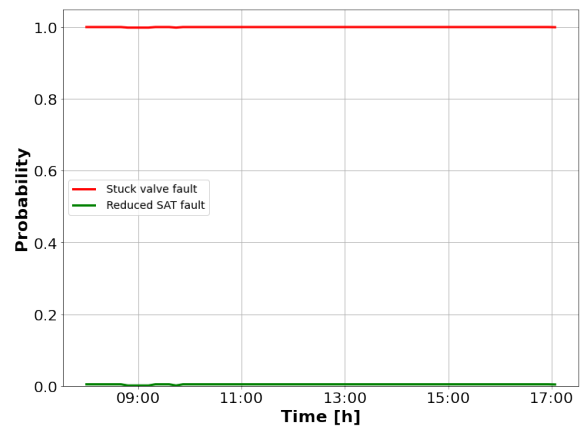
Figure 38: Posterior probability for 50% stuck valve for north cooling coil in office building Breda

Figure 39: Posterior probability for reduced SAT by 2K for north cooling coil in office building Breda
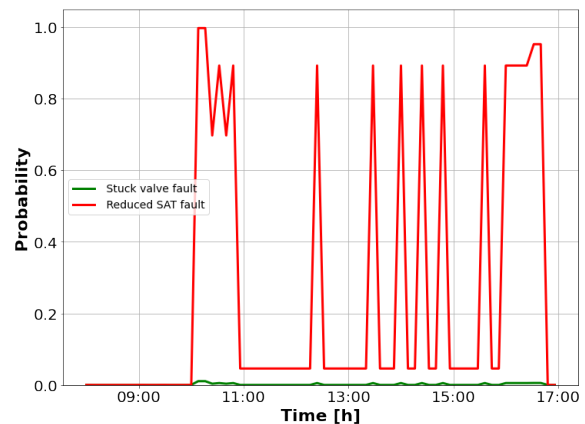

Figure 40: Posterior probability for reduced SAT by 1K for north cooling coil in office building Breda

The developed FDD algorithms which include the two XGBoost models and the DBN, show the ability to detect the low ΔT syndrome and diagnose certain faults which lead to it, which include the stuck

valve fault (50% and 75% stuck) and the reduced SAT fault (2K). The algorithm is not able to detect the low ΔT syndrome when the SAT was reduced by 1K, and this is because a sufficient increase in MFR was not observed. This is probably because the maximum outdoor air temperature for the specific day the fault was introduced, was lower than 24°C. The same fault was introduced during a day when the outdoor temperature was higher than 25°C, and the fault was detected. The results of this case will be explained in section 5.4 with the figures shown in Appendix A7.

Similar analysis of fault detection and fault diagnosis were also conducted for the school building in Nijmegen for the reduced SAT fault, and the FDD algorithms were able to isolate the fault successfully. The results are shown in Appendix A5.

In the next chapter, the complete product development process is discussed, where the requirements of the stakeholders are analysed, and the final product architecture is explained including the integration of different modules and the FDD algorithms discussed in this chapter.

Selection phase: Product features

4

_____

This chapter provides a step-by-step explanation of the product feature selection process which consists of a stakeholder analysis explained in section 4.1 , requirements analysis discussed in section 4.2, and the product architecture explained in section 4.3. These steps are necessary to get a proper understanding of the requirements of different stakeholders and develop the final product layout accordingly, using the different algorithms tested in the previous chapter.

4.1 Stakeholder analysis

The goal of a stakeholder analysis is to identify the different entities who are involved in the project and group them based on their level of interest, participation and influence in the project. The analysis helps in determining the best way to communicate with the different stakeholders and how to involve them in the product development process. A Mendlow's matrix is useful to understand the respective power and interest of different stakeholders within a project such that they can be managed effectively. The different stakeholders of this project are laid out on a Mendlow's matrix as shown in Figure 41.
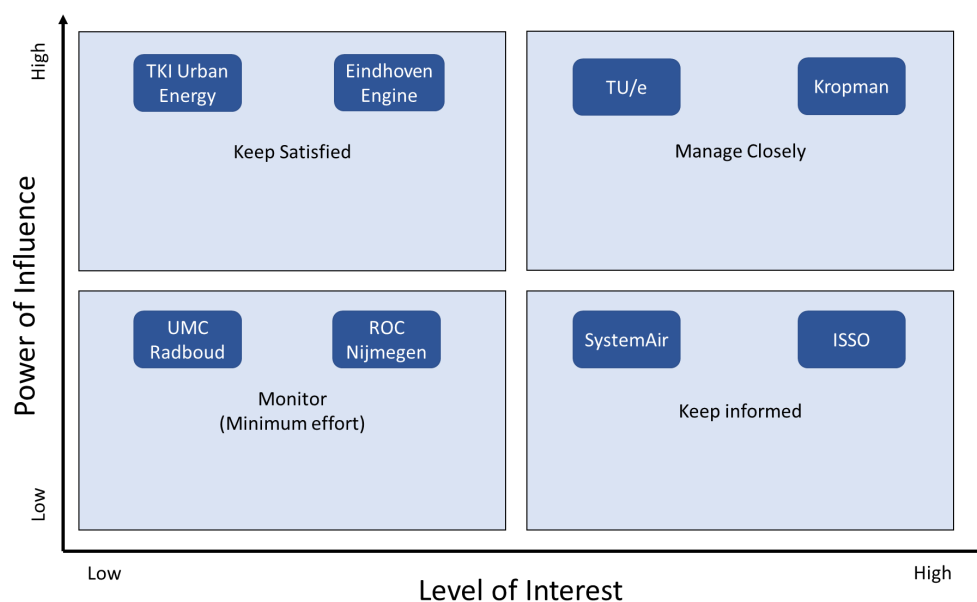
Figure 41: Mendlow's matrix used for the stakeholder analysis

With an understanding of how the different stakeholders need to be managed, the next step is to understand the requirements of each of the stakeholders. This is done by developing user stories, which is a common practice in agile software development (Dalpiaz & Brinkkemper, 2018) . User stories help in capturing the requirements of the stakeholders more precisely in the form of needs/goals. Table 5 shows a list of the different stakeholders, the different roles within a stakeholder organization, and their specific user stories. Alongside each stakeholder, a code is also specified which is used later

to link the defined requirements with the stakeholder. The goals specified in the form of user stories would later be translated into requirements.

Table 5: Stakeholder user stories

Stakeholder	Role	I want <goal>	So that <reason>
Kropman Installatietechniek (A)	HVAC Advisor	To **predict** the low ΔT syndrome at an early stage and identify its possible causes	Energy efficiency of the system can be increased, occupant comfort can be improved, and extreme system failure can be avoided.
		To **get insights** about fault severity levels	I can prioritise my efforts on more serious issues in the system which require immediate attention
		To **receive fault explanations** and to-do actions	I get a preliminary idea of what the issue is, and it saves me time to try and diagnose the issue myself
		Visualizations to explain the fault in the form of heat maps, scatter plots	So that I can understand the issue better and observe the phenomenon from data
	Remote Services Engineer	To **monitor** the energy performance of all my projects with minimum effort in one platform	I don't have to individually monitor energy efficiency issues for each system in a project.
		To **see alarms** from any building site/project immediately	I can inform service engineers to take corrective actions accordingly
		To **receive** detailed information about the fault alarm	I can send the required field technician
	Technician	To **be informed** of corrective actions to be taken in case a fault occurs	I don't have to diagnose the fault myself and eventually saves me time
		To **see** the historical alarms list	I can get an understanding of the system performance and past issues
	Software Development Department Head	An FDD tool which is **modular** in nature and can be integrated to the existing BMS	Iterative and constant improvements and upgrades can be made
	Building Automation Director	To **improve** the current fault detection and diagnosis approach from rules-based to AI-based including predictive maintenance	The value proposition of the tool increases and is in line with market growth
TU/e (B)	University Researcher	To **implement the latest research** in FDD to a commercial product	Buildings are equipped with intelligent systems to achieve energy efficiency
ROC Nijmegen (C)	Building Manager	To **receive automated diagnosis results** for faults in the system	I can take corrective actions quickly and ensure comfortable occupant environment

		To **understand** the impact of faults on important key performance indicators (KPIs)	Decisive actions can be taken accordingly
UMC Radboud (D)	Building Manager	To **monitor** the performance of all AHUs in the university complex and receive alarms on priority basis	It saves me time and makes it easier to isolate faults in specific areas
SystemAir (E)	Product Manufacturer	To **understand** the low ΔT syndrome better	I can provide my clients with a better selection of cooling coils
ISSO (F)	Research institute	To **provide** the Dutch building sector with the **knowledge of better design**	The buildings can be more energy efficient
TKI Urban Energy (G)	Funding organization	To **promote** the development of **intelligent FDD systems** in buildings	There is an increase in the energy performance and energy savings of the building.
Eindhoven Engine (H)	Funding organisation	To **promote innovative research**, collaborate with other researchers and learn from them	There is accelerated innovation to provide technology-based solutions to societal problems.

4.2 Requirements analysis

Based on the user stories of the different stakeholders, a list of functional and realization requirements was generated as shown in Table 6 and Table 7. The product was developed based on the defined functional and realization requirements.

Table 6: Functional requirements

ID	Functional requirements	Concerned stakeholders
F1	The tool should be able to detect multiple faults which can occur simultaneously.	A,B
F2	The tool should be able to provide additional information for faults diagnosed.	A,B,C
F3	Fault detection model training should be completely automated.	A,B,C,D
F4	The tool should be able to provide to-do actions in case faults are diagnosed.	A
F5	The tool should be able to show the severity level of the fault.	A
F6	The tool should be able to show the effect of a fault on key performance indicators.	A,B
F7	The tool should provide the capability to analyse and understand the symptoms in detail.	A,B
F8	The tool should provide the possibility to analyse the fault probabilities for all states of each fault.	
F9	The tool should have the capability to track, monitor and compare the performance of different fault detection models.	A,B

Table 7: Realisation requirements

ID	Realization requirements	Concerned stakeholders
R1	The tool should have modular components.	A,B
R2	The tool should be developed on an open-source platform.	A,B,G,H
R3	The tool should be interoperable with all kinds of BMS platforms.	A,B,C,D
R4	The tool should be generalizable and usable for new building installations.	A,B,G
R5	The knowledge needs to be shared with other organizations within the building services community and other research communities.	F,G,H
R6	Additional sensor requirements and installation methods need to be provided to ensure FDD of low ΔT syndrome is possible in new installations.	E,F

4.3 Product architecture

The architecture of the FDD tool consists of the different functional blocks of the tool and how each of them interact with each other. Figure 42 shows the product architecture with the different modules present in it. Each of the modules are explained below. In general, the tool is developed in Python with different object-oriented classes and methods for each of the modules in the tool.
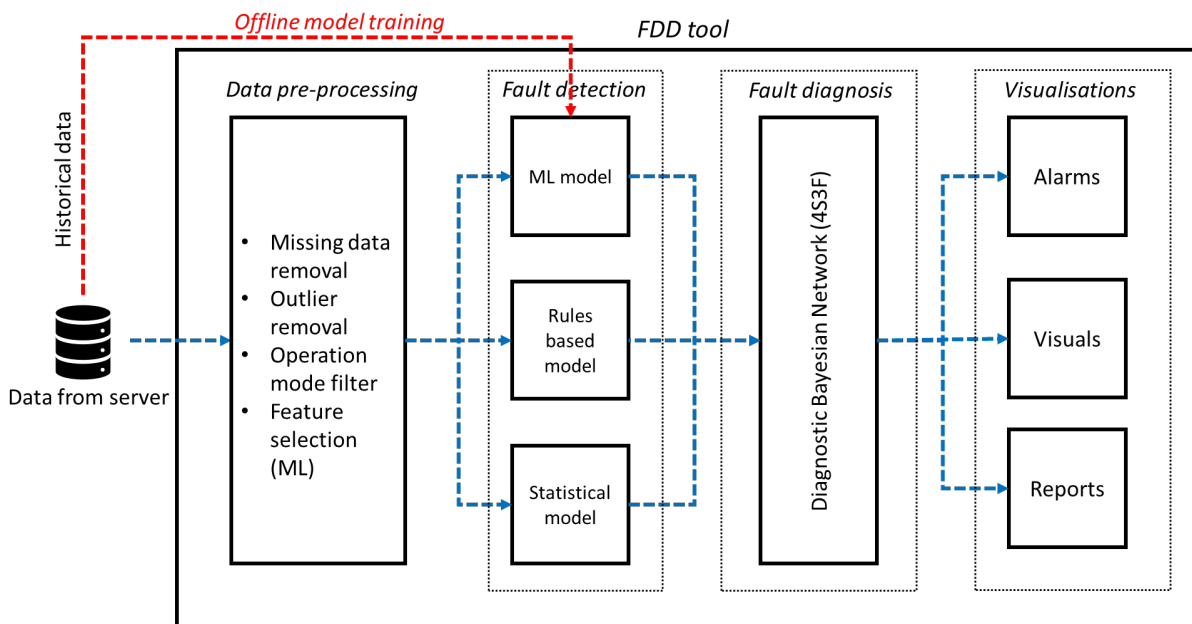
Figure 42: FDD tool product architecture

Data pre-processing

The raw data from the BMS is usually unfiltered, with missing data points and outliers. The first step of the FDD process is to clean the data to ensure that noise and outliers do not influence the FDD process. The pre-processing module consists of different functions like missing data removal, outlier removal and operational mode (cooling mode) filters of the AHU. The missing data fields are removed

rather than interpolated since interpolation could lead to incorrect data points. The operation mode filters consist of simple rules like:

1. CCVP signal > 0%
2. Chiller outlet water temperature <10°C

The filtered data is then passed through the fault detection module.

Fault detection

The fault detection module consists of an offline model development class and model prediction class for the ML models. Each of the classes contain functions to train, predict and evaluate the ML model. The cleaned data is passed through the fault (symptom) detection module which consists of different models where certain data points pass through specific models including ML models, statistical model and simple rules-based models where residuals are generated.

Figure 43 shows an example of the detection of a fault in the air speed using a statistical method known as exponential weighted moving average (EWMA). The predicted air speed which is approximately 2.3 m/s (obtained from the historical moving average) is greater than the measured value by 0.6 m/s, hence indicating a lower airflow.
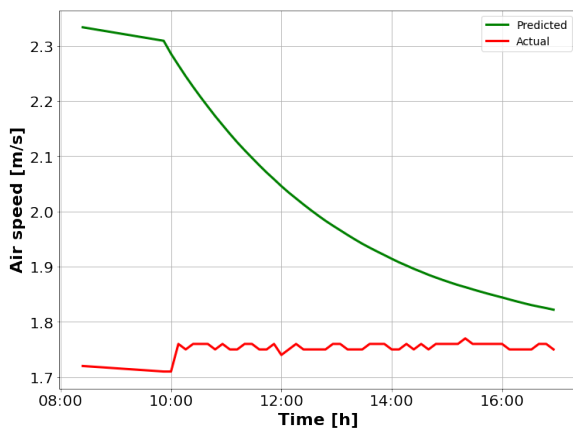
Figure 43: Fault detection of air speed in the AHU using an EWMA statistical model
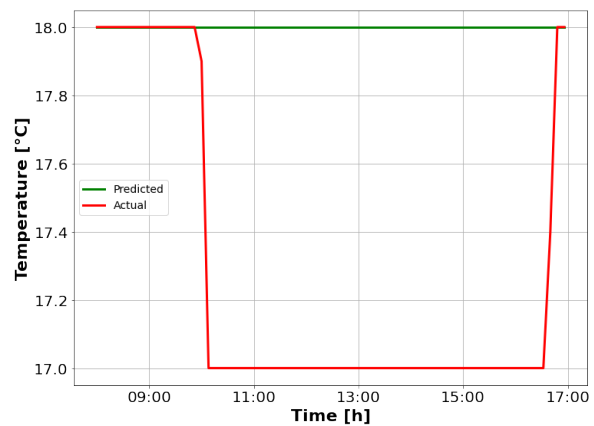
Figure 44: Fault detection of the difference in SAT setpoint and design value using a rules-based model

Figure 44 shows a rule-based symptom detection method where the SAT setpoint is compared with the minimum design SAT setpoint which is a linear equation based on the outdoor air temperature. The fault is detected between 10 am and 4:30 pm, as seen in the figure.

Figure 45 shows the detection of a symptom using an ML model. The model is used to predict the fault-free CCVP and is compared with the measured faulty CCVP. The ML models are trained and developed using historical data from the server. This process will be done only in the beginning and when model performance starts to drop below certain thresholds. Once the residuals are generated from each fault detection model, the data is sent to the fault diagnosis module.
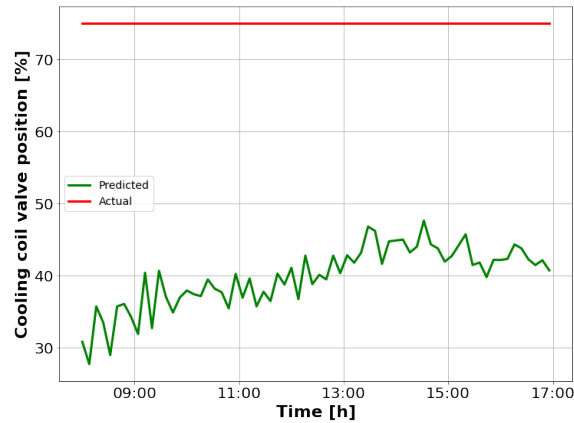
Figure 45: Fault detection of CCVP deviation using an ML model

Fault diagnosis

The fault diagnosis module consists of a DBN, which is based on the 4S3F method. The DBN consists of symptom nodes, which is where all the data from the BMS is fed into, and it contains fault nodes which is where the posterior probability for each fault is calculated. The symptoms nodes receive residuals from each of the fault (symptom) detection models and are translated to symptom states which are readable by the DBN. The DBN then calculates posterior probabilities for the different fault nodes present and maps the faults with the underlying symptoms.

Visualizations

The data from the fault diagnosis module is transferred to the visualisations module where the data is represented in the form of different visuals (bar graphs, line graphs and heat maps), reports (interactive tables), and alarms (coloured indicators to signal the presence of low ΔT syndrome). Additional visualizations of the DBN are also present which show the connection between the faults and where they are located in the P&ID diagram.

The next chapter discusses the final product development process including prototyping, realisation of the final tool, verification, validation, and evaluation.

Shaping phase: Product development

<div align="right">

5

</div>

_____

This chapter discusses the product development, product testing and evaluation of the developed FDD tool. Section 5.1 discusses the prototyping of the tool, where the product layout is designed based on the agile software development process. The final developed product is discussed in section 5.2, where the different features of the tool are highlighted and explained. Section 5.3 presents the verification of the tool, where the functional requirements of the tool are checked for by analysing the different features of the tool. Section 5.4 presents the validation of the tool where data from the in-situ experiments conducted at the office building in Breda were analysed by the FDD tool. The tool's ability to perform based on the specified requirements is analysed. Section 5.5 presents an evaluation of the tool in terms of its ability to detect and diagnose all the faults, and its limitations.

5.1 Prototyping

The main aspect of the FDD tool where the user interaction occurs is the dashboard (visualization), where the user interface (UI) and user experience (UX) aspects need to be considered. The product dashboard was developed based on the agile software development method, a flow schema of which is shown in Figure 46. The agile development method includes requirements discovery and solutions improvement through the collaborative effort with the end user (Dalpiaz & Brinkkemper, 2018). The main purpose of following such a method was to implement practices such as evolutionary development, continual improvement, flexible responses to changes in requirements and a better understanding of the problem to be solved.

To get a better understanding of what the different stakeholders expect, an initial design prototype was developed on the prototyping platform called Figma. The design prototype was made based on the initial requirements which have been laid out.

Figure 46: Agile software development method

The first prototype was developed in Figma, which is a web-based vector graphics and prototyping tool which is mainly used by designers for building digital products. The first design prototype was shown to the stakeholders to get their feedback and impressions of how the tool would look like and what expectations they have in terms of features. Their feedback, expectations, and suggestions were considered while updating the prototype layout. A second update of the prototype tool was made to get a final confirmation of the dashboard layout. The different screens of the design prototype developed in Figma are shown in Appendix A6.

Once the design was confirmed, the actual frontend of the product was developed in Dash, which is a python-based dashboarding application from the Plotly library. The application (frontend) developed in Dash is rendered in the web browser using an HTML template with CSS and JavaScript elements references inside it. The backend of the tool includes algorithms which were developed using the specific libraries required for fault detection (sci-kit learn) and fault diagnosis (pomegranate). Pomegranate is an open-source probabilistic modelling package in Python (Schreiber, 2018) used specifically for Bayesian network modelling for the FDD tool. The prototype of the tool was made and shown to stakeholders so that they can see how the tool works. Their feedback and suggestions were considered, and the product was updated accordingly.

5.2 Final product

The final application was developed which can be deployed as a standalone application as well. Figure 47 to Figure 55 illustrate the different screens of the final product developed in Dash. Figure 47 shows the main screen of the application with the different projects (office building in Breda and school building in Nijmegen) along with their locations on the map. A purple icon is placed on the specific coordinates of each project (indicated by '**a**' in the figure), where hovering over it provides information about the number of alarms present in the building. The size of the icon is dependent on the number of existing alarms. Therefore, the larger the icon, the more fault alarms present. To access the main alarms page of a particular building, the black button on the building image (indicated by '**b**' in the figure) needs to be clicked on. The main menu of the tool can be accessed by click on the "*Explore*" button (indicated by '**c**' in the figure).
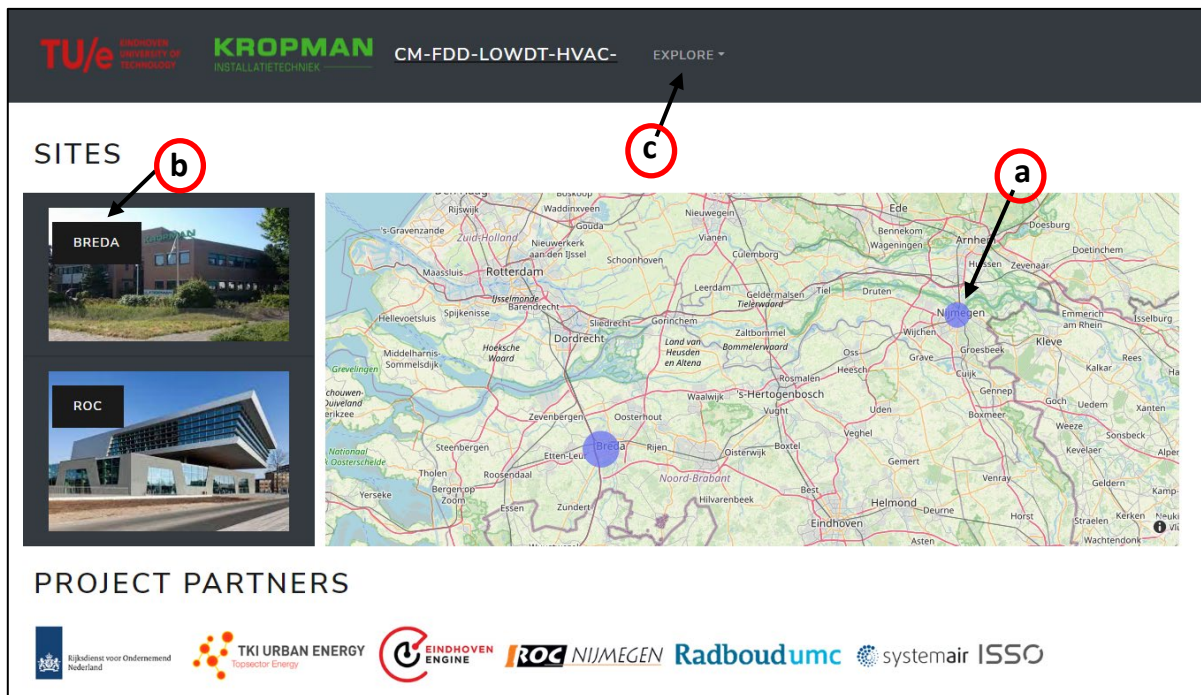
Figure 47: FDD tool main screen
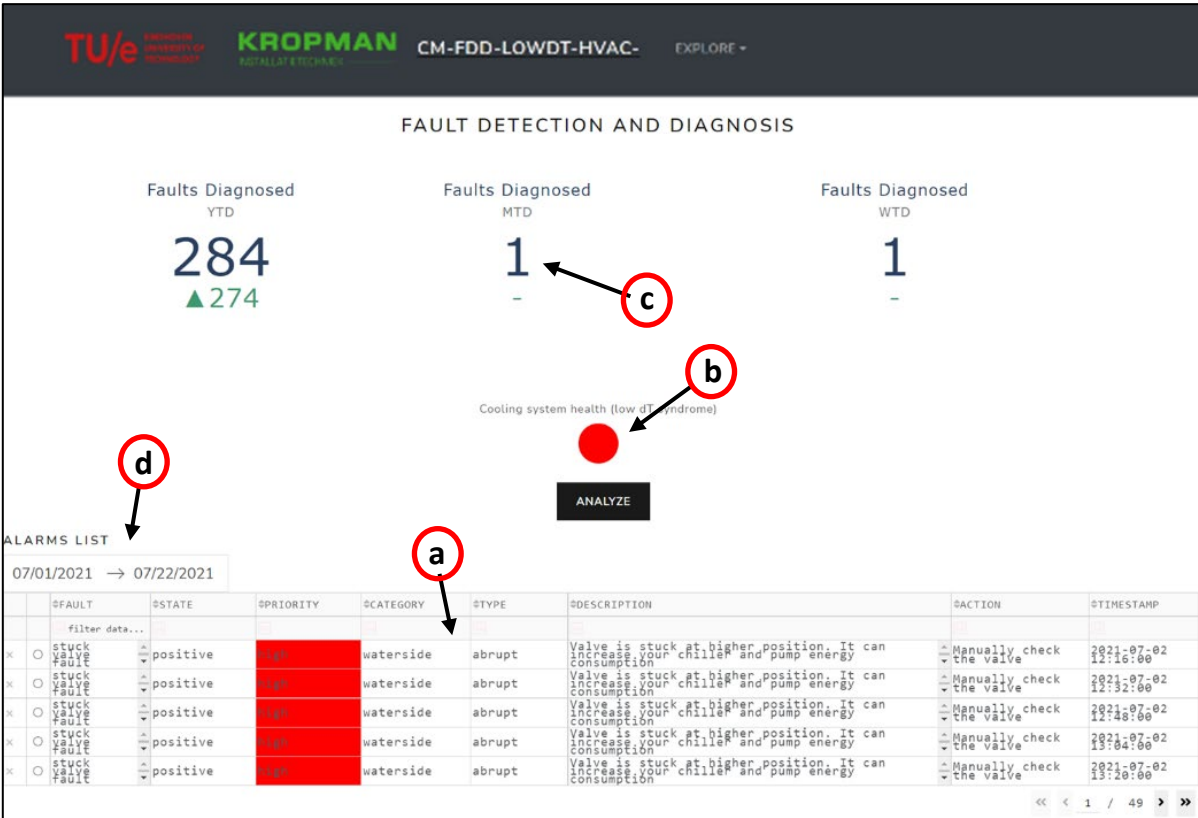
Figure 48: FDD tool alarms screen for a facility manager

The main alarms page for the building site is shown in Figure 48, where the diagnosed fault, its priority, description and consequent to-do actions for the particular fault, are displayed in an interactive alarms table (indicated by '**a**' in the figure). The page also includes indicators (indicated by '**c**' in the figure) for other general information including the total number of faults diagnosed for the year, month, and week.
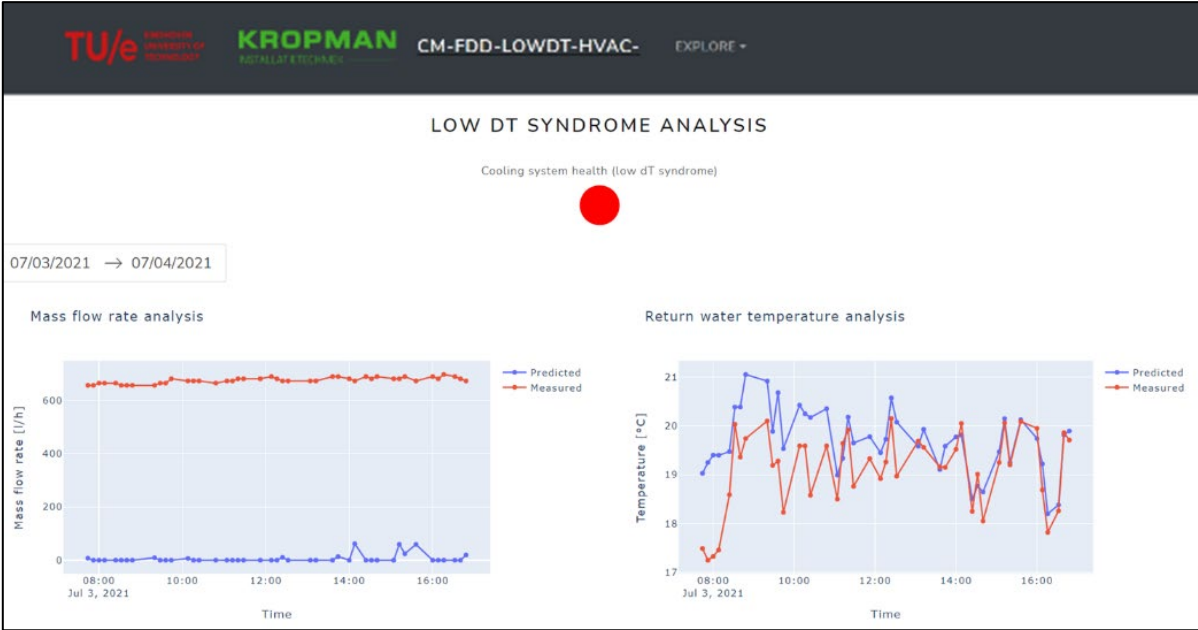
Figure 49: FDD tool low ΔT syndrome analysis screen for an HVAC expert

There also includes a round low ΔT syndrome status indicator (indicated by '**b**' in the figure) which turns red in colour if the low ΔT syndrome is present and turns green when the system is functioning normally without the low ΔT syndrome. A date selector is also present (indicated by '**d**' in the figure) to surf freely through the different days of the year. The low ΔT indicator and the alarms table get updated accordingly. On further clicking the "**Analyse**" button below the indicator, a low ΔT syndrome analysis page appears (Figure 49), where the MFR and RWT trends can be studied in more detail. This page is intended for the HVAC expert who would like to study the low ΔT syndrome in detail by looking at trends of MFR and RWT. As seen in Figure 49, the graphs on the screen include predicted and measured values, a date selector, and a low ΔT syndrome indicator.
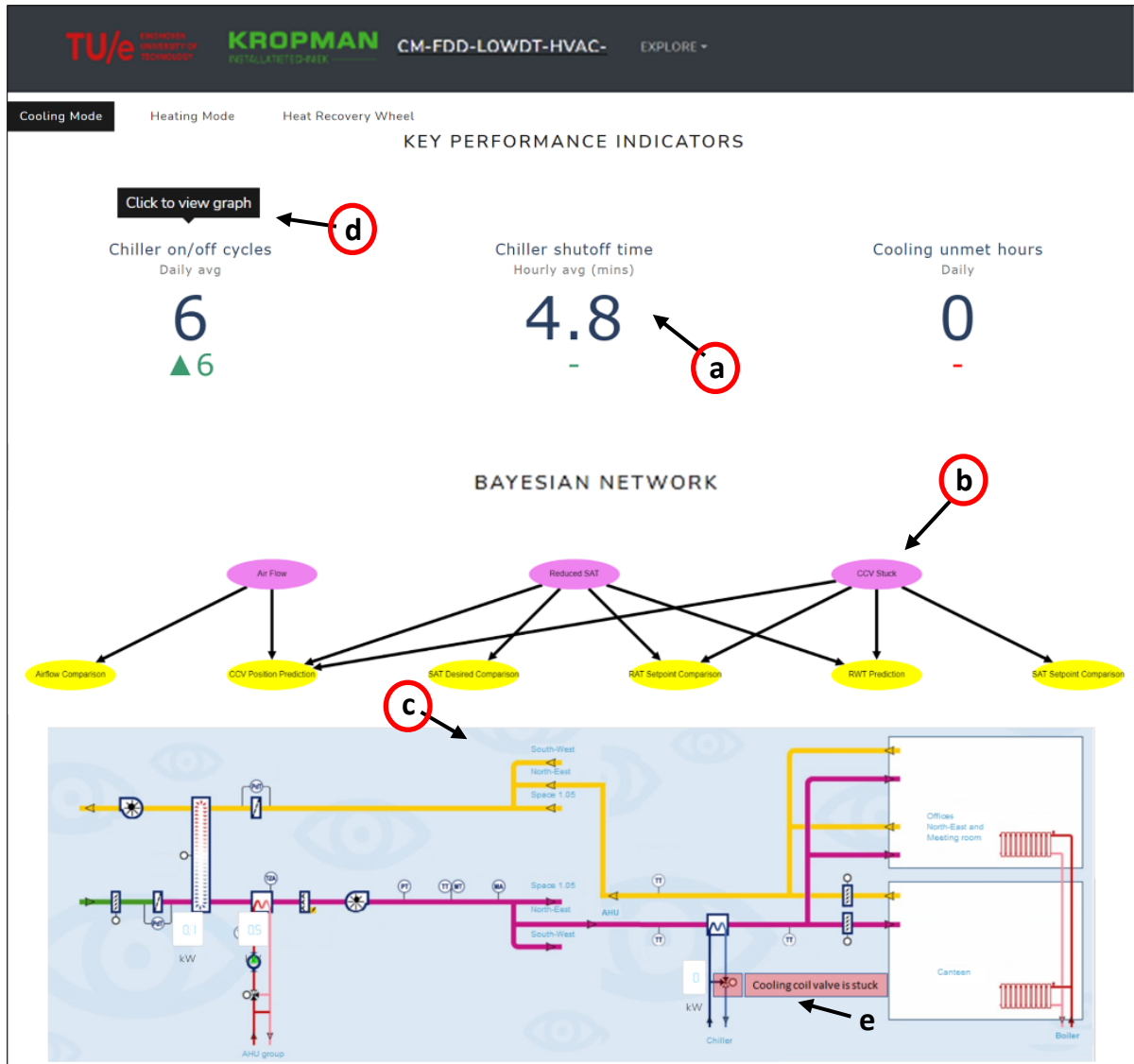
Figure 50: FDD tool DBN screen for an HVAC expert

The DBN screen can be accessed by clicking on the "**Explore**" button and selecting the specific DBN for the required site. Figure 50 shows the DBN screen for the north cooling coil of the office building in Breda where the KPIs (Key performance indicators) like chiller on/off cycles, unmet cooling hours, and average chiller shutoff time, (indicated by '**a**' in the figure), the DBN (indicated by '**b**' in the figure), and the P&ID diagram) (indicated by '**c**' in the figure), are displayed. The specific KPIs are used to

understand the impact of the faults on energy consumption (reflected upon the number of chiller on/off cycles and average chiller shutoff time) and comfort (unmet cooling hours). On clicking any of the KPIs, a modal appears where the user can conduct further analysis. A tooltip also appears when hovering over the KPI (indicated by '**d**' in the figure), to indicate that a graph will be displayed if clicked on. Additional user-guiding features include highlighting of the fault in the P&ID diagram (indicated by '**e**' in the figure), when hovering over a specific fault node in the DBN. In the figure, the CCV stuck fault node is hovered over, to highlight the pink box over the cooling coil valve. On clicking the KPIs (indicated by '**d**' in the figure), a modal opens which gives a more detailed analysis of the KPIs.

Figure 51 shows a multi-tab modal for the chiller staging performance, which appears after clicking the chiller on/off cycle KPI in Figure 50. The modal consists of an interactive graph with multiple tabs (indicated by '**a**' in the figure) to assess both total chiller staging cycles in the form of a bar plot, as well as chiller staging cycles as a line plot. The graph also includes a range slider for outdoor air temperature (indicated by '**b**' in the figure), to display datapoints that lie in the specific range selected.
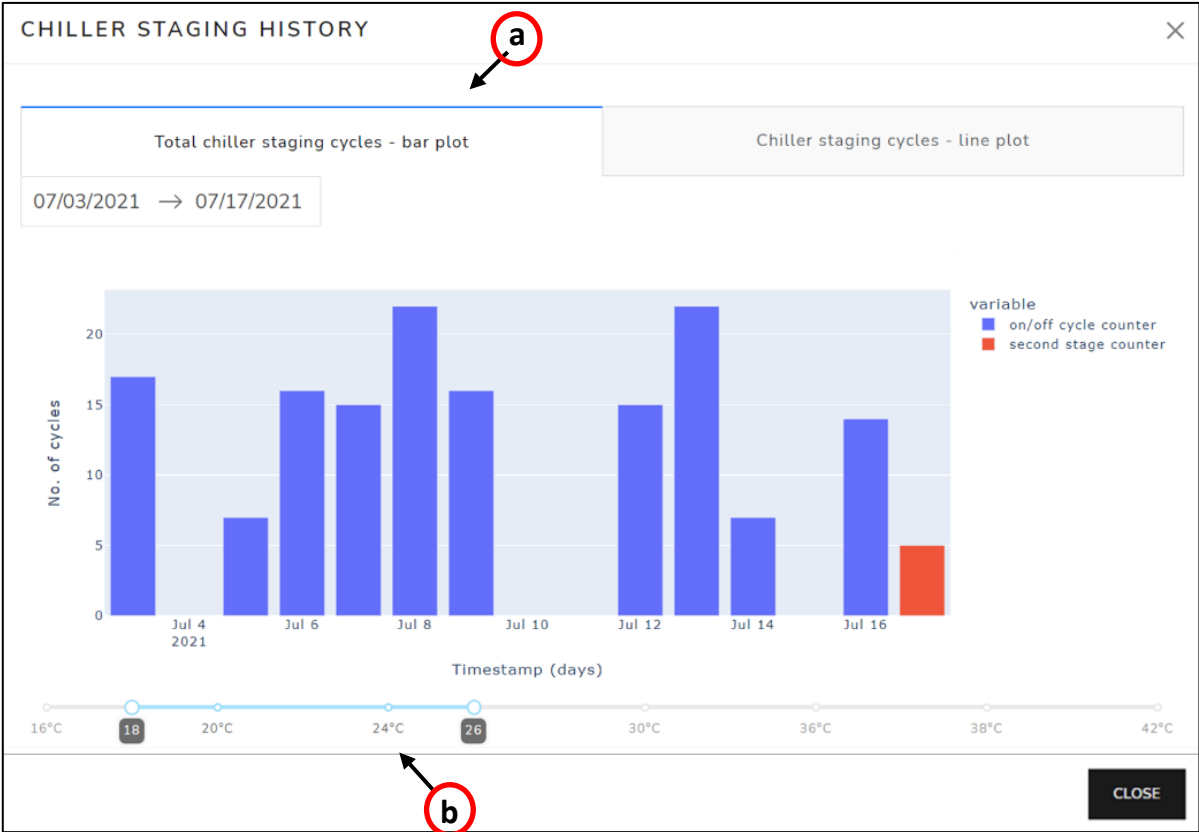
Figure 51: FDD tool KPI modal for an HVAC expert

On clicking the specific fault and symptom nodes on the DBN, Figure 52 and Figure 53 open as modals respectively. Figure 52 shows the fault probabilities in the form of a line graph, where the fault probabilities for different states of all the symptom nodes are available for analysis. The different states of the fault nodes can be checked using the check boxes (indicated by '**a**' in the figure), to display the required probability values. The graphs are interactable in the manner that they can be zoomed into and analysed for a particular period.
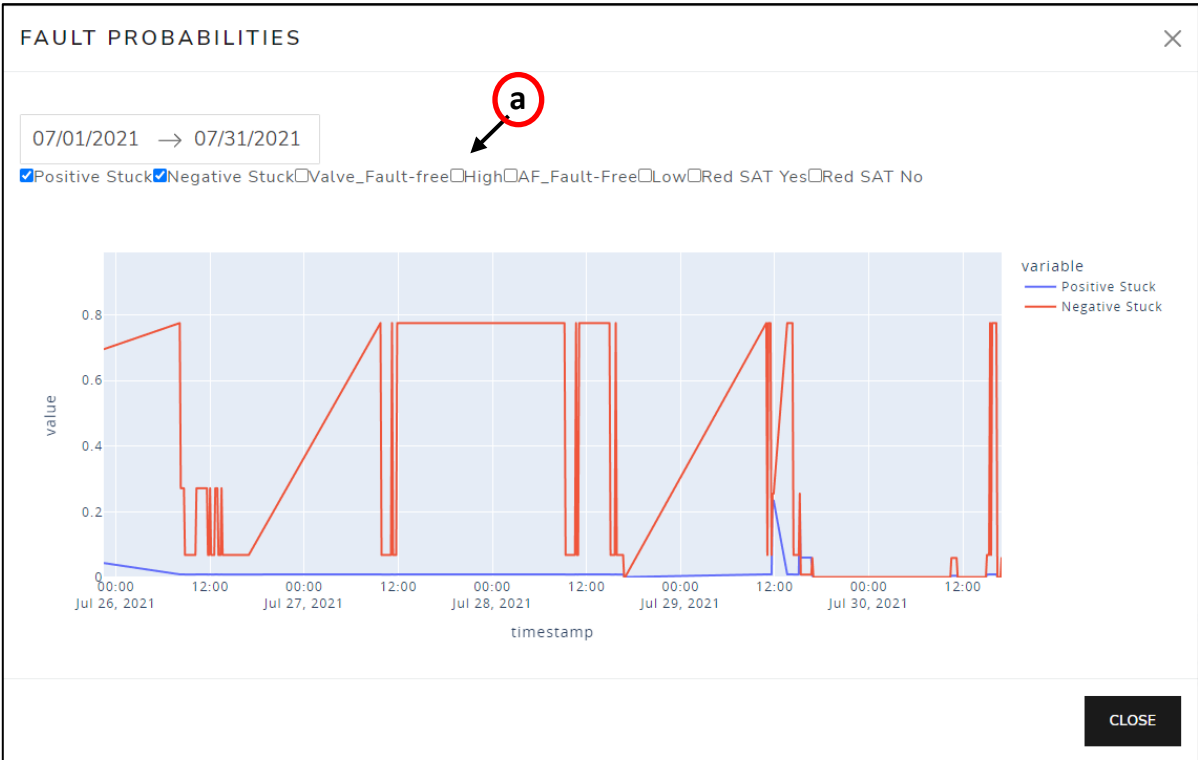
Figure 52: FDD tool fault probability modal for an HVAC expert

Figure 53: FDD tool symptom residual analysis modal for an HVAC expert

Figure 53 shows the modal where the intensity of the residuals of the symptom nodes are presented in the form of a heatmap. A user can toggle through different symptom nodes by just clicking on the

different tabs (indicated by '**a**' in the figure). The data can be accessed by selecting the specific year and week of interest (indicated by '**b**' in the figure).

Figure 54 shows the ML screen which displays the performance of the different ML models. Additional information on the models is also displayed including the RMSE in the form of a gauge (indicated by '**a**' in the figure), R^2 score in the form of a gauge (indicated by '**b**' in the figure) and the number of features (indicated by '**c**' in the figure). An important feature is the display of a message which shows whether model re-training is required (indicated by '**d**' in the figure). The historical performance of the model can be analysed by clicking on the "***Check performance***" button (indicated by '**e**' in the figure).
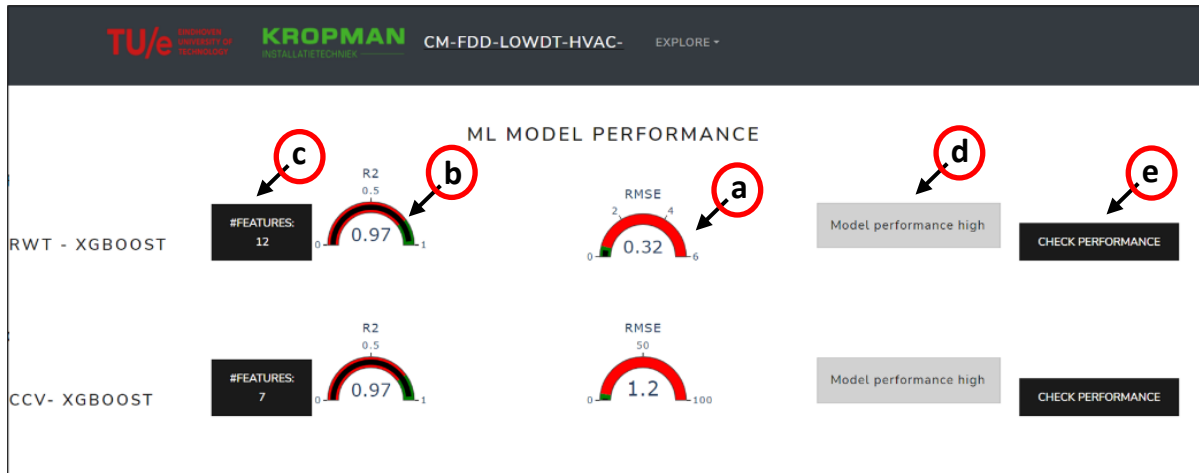
Figure 54: FDD tool ML screen for an ML expert

Figure 55 shows a modal that displays the historical performance of the ML model in terms of R^2 score and RMSE in the form of an interactable line graph. The historical scores can be analysed to check if the model is performing sufficiently well within the benchmark values.

Figure 55: FDD tool ML analysis modal for an ML expert

42

With the final product developed, the next step is to verify the product and see if it meets all the requirements of the stakeholders. This is done by analysing the individual features of the product and checking if it satisfies the requirements specified in Table 6.

5.3 Verification

As a part of the product evaluation, verification is conducted to ensure that the product meets all the requirements set by the stakeholder. This is done by evaluating the different features present in the tool, without actually running the application.

The different functional and realisation requirements from Table 6 and Table 7 respectively, are analysed in detail, and the relevant properties or features of the tool are highlighted next to each requirement as shown in Table 8 and Table 9.

Table 8: Verification of functional requirements

ID	Functional requirements	FDD tool feature developed
F1	The tool should be able to detect multiple faults which can occur simultaneously.	The algorithm of a DBN allows multiple faults to be diagnosed at the same time since it is based on probabilistic inference.
F2	The tool should be able to provide additional information for faults diagnosed.	The alarms screen of the FDD tool provides information about the specific timestamp at which the fault was diagnosed, the priority level and the type of fault.
F3	Fault detection model training should be completely automated.	The FDD tool consists of a module that tracks the performance of the ML models and automatically re-trains them based on the RMSE and R^2 score.
F4	The tool should be able to provide to-do actions in case faults are diagnosed.	The alarms screen of the FDD tool provides to-do actions for each fault diagnosed by the DBN.
F5	The tool should be able to show the priority level of the fault.	The alarms screen of the FDD tool provides a colour-based indicator of the specific fault where red means high priority, orange means low priority.
F6	The tool should be able to show the effect of a fault on key performance indicators.	In the DBN screen of the tool, different key performance indicators are presented in the form of icons with indicators of the average trend. The icons are also clickable, which show interactable graphs with the possibility to analyse in detail.
F7	The tool should provide the capability to analyse and understand the symptoms in detail.	In the DBN screen of the tool, on clicking the symptom nodes of the DBN, a modal appears which provides a heatmap of the residuals of all the symptom nodes.
F8	The tool should provide the possibility to analyse the fault probabilities for all states of each fault.	In the DBN screen of the tool, on clicking the fault nodes of the DBN, a modal appears which provides a line graph displaying the fault probabilities of all the states of each fault.
F9	The tool should have the capability to track, monitor and compare the performance of different fault detection models.	In the ML screen of the tool, each ML model is listed in a row format where the performance metrics are displayed in the form of simple gauges. A message box is also included which informs the user if the relevant ML model requires re-training.

Table 9: Verification of realisation requirements

ID	Realization requirements	FDD tool feature developed
R1	The tool should have modular components.	The back-end of the tool consists of multiple python pages for each screen with modular blocks of code which can be replicated for other projects.
R2	The tool should be developed on an open-source platform.	The tool is developed using the python programming language with open-source components including ML libraries, DBN library, and dashboarding library.
R3	The tool should be interoperable with all kinds of BMS platforms.	The tool communicates with the data server over API. Communication over API is possible for all kinds of BMS as long as a history server for the specific building exists.
R4	The tool should be generalizable and usable for new building installations.	The tool consists of modular components and a structure where new projects can be easily added.
R5	The knowledge needs to be shared with other organizations within the building services community and other research communities.	The tool is developed as a private GitHub project where access can be provided to relevant organizations and research partners. Knowledge sharing sessions of the developed tool are also organized among the building services community in the Netherlands.
R6	Additional sensor requirements and installation methods need to be provided to ensure FDD of low ΔT syndrome is possible in new installations.	An instruction manual for the FDD tool is developed, providing users with information about the features of the tool and the pre-requisites for proper functioning of the FDD algorithms. The sensor requirement list for a minimum viable product is also listed.

5.4 Validation

With the completion of the verification of the tool in terms of all the requirements, the next step is to validate the tool with a live in-situ test. The tool was tested in the office building in Breda, where the initial algorithm testing study was conducted as discussed in chapter 3. The stuck valve fault (50% stuck) and the reduced SAT fault (1K and 2K) were introduced to the north cooling coil, in May 2022 and June 2022. For the stuck valve fault, the setpoints for the valve position were fixed in the BMS whereas for the reduced SAT fault, the minimum temperature setpoints were reduced by 1K and 2K. Table 10 shows the specific dates and timings during which the faults were introduced.

Table 10: List of faults introduced to the north cooling coil at office building Breda for validation of the tool

Date	Time	Fault	Severity level
20-05-2022 to 22-05-2022	15:30 – 22:40	Stuck valve	50%
03-06-2022	08:15 – 12:00, 15:30 – 17:00	Reduced SAT	2K
15-06-2022	12:30 – 17:00	Reduced SAT	1K
17-06-2022	12:40 – 17:00	Reduced SAT	2K

The validation of the product was done by analysing the alarms page and DBN page of the FDD tool during the days when the different faults were introduced. Figure 56 shows a screenshot of the alarms page during the period when the 50% stuck valve was introduced. The low ΔT syndrome indicator blinks red which shows that the low ΔT syndrome has been detected in the system. The alarms list at the bottom of the figure shows the fault diagnosed as the stuck valve fault with a description of the possible issue and a to-do action required to solve the problem.
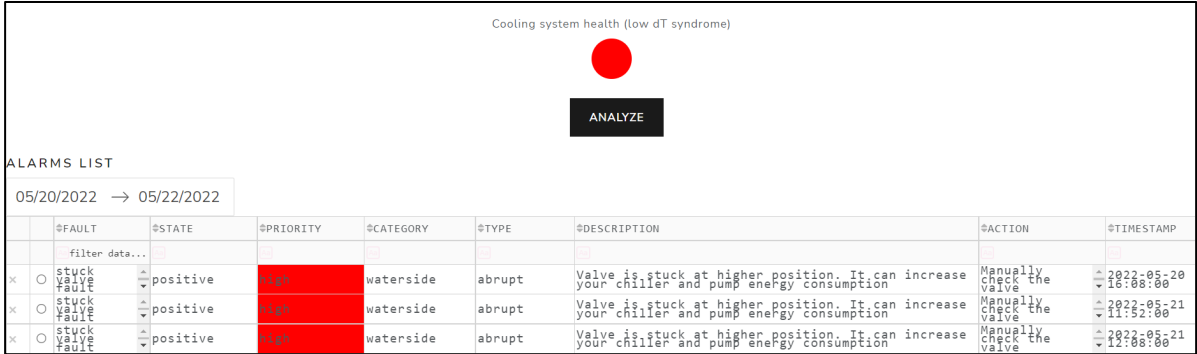
Figure 56: Screenshot of alarms page for validation of 50% stuck valve fault

Figure 57 shows the probability distribution modal for all possible faults which can be diagnosed by the DBN. It is seen that the probability of the stuck valve fault at the positive state is almost equal to 1 between the evening of 20th June and the evening of 22nd June, whereas the probability values for the other faults are close to 0.
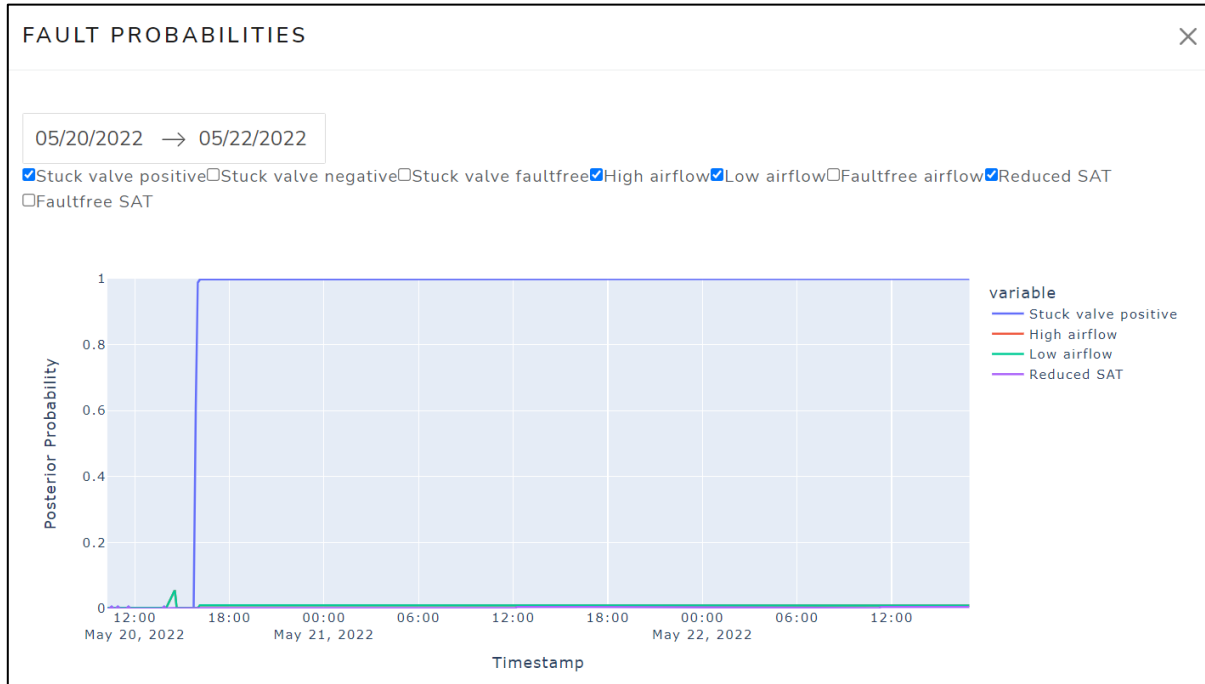
Figure 57: Screenshot of DBN page modal for validation of 50% stuck valve fault

The validation of the tool for the other three fault cases was also conducted and the screenshot results are shown in Appendix A7. All the three faults were diagnosed by the DBN, including the reduced SAT by 1K fault. This was probably because the reduced SAT faults were conducted during warm days when

the outdoor air dry-bulb temperatures rose above 24 °C. Since the cooling demand was high during the 3rd, 15th, and 17th of June 2022, the introduction of the fault resulted in the occurrence of the low ΔT syndrome as there was sufficient MFR increase and RWT decrease in the cooling coils. This is logical, since the SAT would be reduced only when the outside temperature is too hot, and the indoor conditions get too uncomfortable.

The validation of the tool for the school building in Nijmegen was also conducted, with the different screens for the fault cases shown in Appendix A8. No additional fault experiments were conducted for validation apart from the experiments of summer 2020. It was observed that the tool was able to detect the low ΔT syndrome and diagnose the reduced SAT faults which caused it.

5.5 Evaluation

With the validation of the FDD tool completed, it is necessary to analyse the tool in terms of its fault classification accuracy and other limitations. The FDD tool in general was able to detect the low ΔT syndrome in almost all the fault cases except for some of the reduced SAT faults when the outside air temperature was not very high. There is no clear explanation as to why this happens since the prediction is black-box. The most likely assumption is the influence of a lower outdoor air temperature.

Figure 58 shows the confusion matrix obtained for the DBN analysis conducted for the validation between 3rd March 2022 and 17th June 2022. During this period 4 fault cases were introduced, as explained in section 5.4. The classification accuracy of the DBN is usually described by performance indicators like accuracy ($\frac{TP+TN}{TP+TN+FP+FN}$), sensitivity ($\frac{TP}{TP+FN}$), specificity ($\frac{TN}{TN+FP}$) and precision ($\frac{TP}{TP+FP}$) (Ruuska et al., 2018), where TP, FP, TN and FN are number of true positives, false positives, true negatives and false negatives respectively.
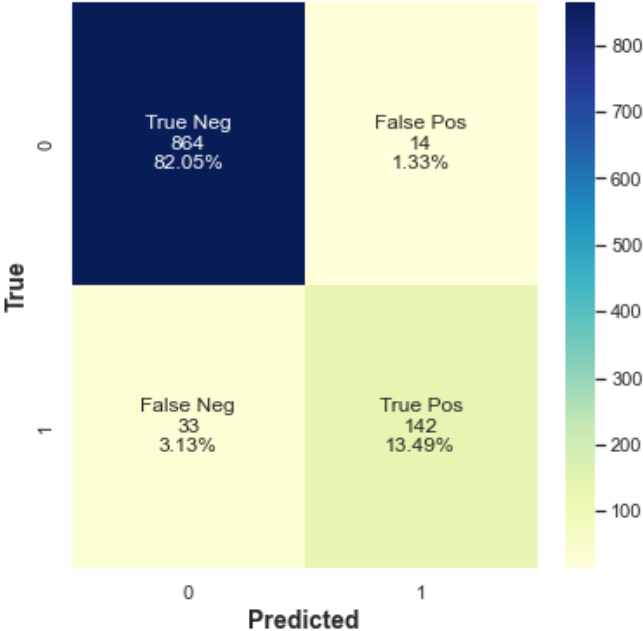
Figure 58: Confusion matrix for the DBN for the north cooling coil of the office building in Breda

The DBN was able to detect the low ΔT syndrome with an accuracy of 95.5%. The sensitivity of the DBN, which describes the ability to positively detect the fault, was calculated to be 0.81. The specificity

of the DBN, which describes the ability to avoid false alarms, was 0.98. The precision of the tool, which describes the relevant positively classified instances, was 0.91. The fault diagnosis accuracy (accuracy of labelling the faults that caused the low ΔT syndrome) of the tool was calculated to be 93%.

False positive cases were identified for certain timestamps when the ML models made incorrect predictions leading to a fault being diagnosed when it really isn't present. Even though the DBN provides a high probability for the fault, it is not usually reported in the alarms table since the fault needs to be detected for consecutive timesteps over a certain window period.

An incorrect fault labelling was also identified for the reduced SAT by 2K fault introduced on the 17th of June 2022. The fault was diagnosed as a reduced SAT fault for the first 30 minutes but was later diagnosed as a positive stuck valve fault. This is because the outdoor air temperature gradually increased from 26 °C to 30 °C during the period when the fault was introduced. At this point, the design SAT setpoint was calculated to be approximately 16 °C, the value to which the SAT was initially reduced to. Due to this, the specific symptom node to compare the design SAT with the actual SAT did not detect a deviation and hence the DBN was not able to detect the fault.

Multiple false negatives cases were also identified by the DBN. Apart from the reduced SAT and stuck valve faults, sensor offset faults were also introduced into the system as a part of another research project in the same consortium. These faults were introduced during the month of August 2021 and September 2021. Some of the sensor offset faults that can lead to the low ΔT syndrome include the SAT sensor offset fault (positive offset), the RAT sensor offset fault (positive offset) and the SAT sensor (before cooling coil) offset fault (negative offset).
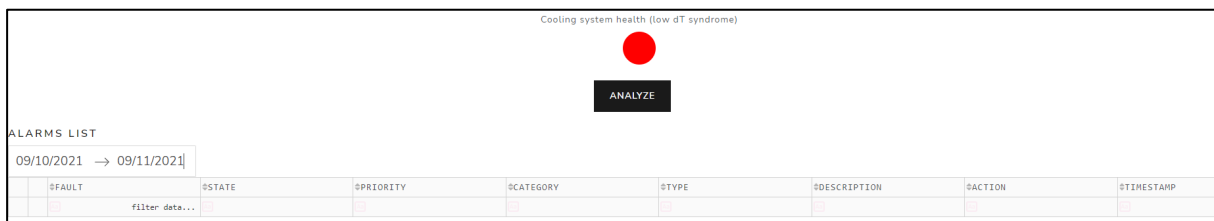
Figure 59: Screenshot of alarms page showing detection of low ΔT syndrome but no diagnosis during the sensor offset faults
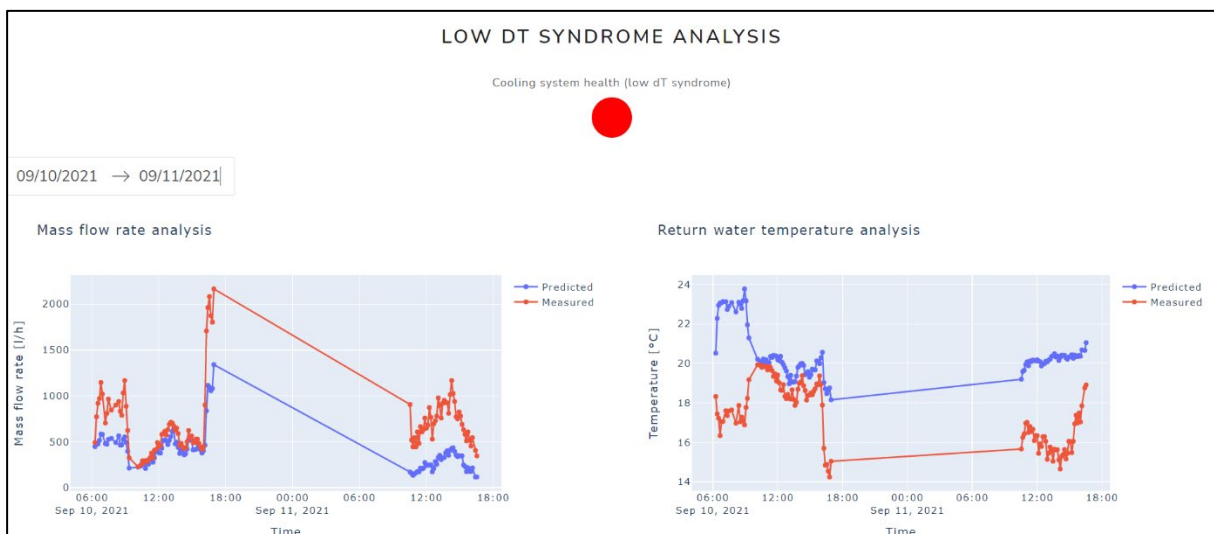
Figure 60: Screenshot of low ΔT syndrome page with MFR and RWT comparison graphs

Figure 59 shows a screenshot of the alarms page for the period when the sensor offset fault was introduced. Between the 10th and 11th of September, an RAT sensor offset (positive offset) fault was introduced into the system with an offset of +3K, but no diagnosis was made as seen in the empty alarms table. Since there was an increase in the MFR and a reduction in the RWT, the low ΔT syndrome was detected, as seen in Figure 60, where the red lines indicate the measured value, and the blue lines indicate the predicted value.

The DBN was not able to provide a suitable diagnosis since there was no specific fault node present for the sensor offset faults. Therefore, in the instance when the low ΔT syndrome is detected, but no alarms are displayed, it is possibly due to the occurrence of other faults which are not included in the fault list of the DBN.

The evaluation of the tool for the school building in Nijmegen was also conducted and it was found that the tool could detect the low ΔT syndrome with an accuracy of 95.8%. The DBN had a sensitivity of 0.94, specificity of 0.96 and a precision of 0.49. The overall fault diagnosis accuracy of the tool was calculated to be 94.8%. The lower precision is since the SAT setpoint, which is calculated using a cooling curve based on the outdoor air temperature, was not properly set for this HVAC system (compared to a proper reference curve as in the building in Breda), therefore giving certain false positive cases for the SAT setpoint comparison symptom node.

The DBN in general clearly shows that it can detect the low ΔT syndrome, providing visual alarms to the user, and providing diagnostic results as well. The overall performance of the tool, it's advantages, limitations and further recommendations for its improvement are discussed in the next chapter.

Discussion

<div align="right">6</div>

_____

In this chapter, the performance of the FDD tool is discussed, highlighting its advantages, drawbacks and limitations.

In this study, the low ΔT syndrome was analysed in detail by conducting fault simulations in EnergyPlus, where the impact of each fault in terms of energy consumption and comfort, and the characteristics of the faults in terms of MFR increase and RWT decrease were studied. The analysis showed that the stuck valve fault and the reduced SAT fault had the highest impact on energy consumption. Therefore, an FDD tool was developed to detect the low ΔT syndrome in the HVAC system and diagnose these specific faults. To make the tool as generalizable as possible, the low ΔT syndrome detection is based on the CCVP and RWT sensor data, which is widely available in most existing HVAC systems. This approach works well for smaller CHW systems with a constant pressure drop across the valve. When the tool is to be implemented for larger CHW systems, where there is a higher possibility of fluctuating pressure drop across the valve, a virtual MFR meter is used to detect the low ΔT syndrome. This ensures reliable alarms for the building operator/technician so that appropriate actions can be taken.

The validation of the tool showed that it can detect the low ΔT syndrome in the system and diagnose two kinds of faults (stuck valve and reduced SAT) that can lead to it. In some instances, the reduced SAT by 1K fault cannot be easily detected by the XGBoost based ML prediction models since there might not always be a substantial increase in the MFR of water or a decrease in the RWT. Other kinds of faults which can lead to the low ΔT syndrome, including sensor offset faults and coil fouling are not included in the DBN. In the instances that such kinds of faults occur in the system, the fault detection model will detect the low ΔT syndrome but not provide any diagnosis result since the relevant fault nodes do not exist. In this situation, the building operator/technician would have to diagnose the fault manually. However, the DBN can be altered to include relevant symptom nodes and fault nodes to expand the fault node list.

The visualization layer of the tool (dashboard) provides the user with information about instances when the low ΔT syndrome was detected and the diagnosed fault along with to-do actions on the main alarms page. Interactable graphs for ML predictions and posterior probability values are also present to assist the user in certain findings. In the situation where the low ΔT syndrome is caused by other faults (for e.g., sensor faults), the low ΔT status indicator would turn red in colour but not show any diagnostic result in the alarms table. This limitation could be further improved by including a suggestion to the user in the alarms table, that a fault was detected but not diagnosed and that manual diagnosis needs to be done.

Conclusion

<div align="right">

7

</div>

This chapter concludes the research and product development work presented in this project along with recommendations for future work.

7.1 Conclusions

An FDD tool has been developed which can detect the low ΔT syndrome and diagnose two kinds of faults which caused it. The tool is developed in Python using the dashboarding framework called Dash, thus making it open-source, generalisable, and scalable. The tool was developed and tested for two case study buildings, where the low ΔT syndrome was detected and diagnosed for the stuck valve and reduced SAT faults.

For the tool to be functional as a minimum viable product, there is no requirement of additional sensors which require complex installation procedures, for e.g., an MFR meter. The fault detection algorithms are based on the RWT and CCVP sensors, which are generally available in most CHW installations. This approach is usually feasible for small and simple CHW systems. In the case of larger buildings with more complex installations, the use of CCVP as a means of detecting the low ΔT syndrome reduces the accuracy of the model since the pressure drop across the valve would not be constant. At this point, the CCVP and MFR do not have the same relationship as in a constant pressure drop condition, leading to inaccurate estimations of MFR demand. In this case, a virtual MFR meter should be used to calculate the fault-free MFR using energy balance equations. Another virtual MFR meter should be developed to measure the actual faulty value using a regression model based on pressure difference of water across the cooling coil. Both faulty and fault-free values of MFR are required to generate residuals and identify if there is an increase or decrease in MFR.

The DBN developed for the office building in Breda was able to diagnose the faults with a diagnosis accuracy of 93% during a validation period of two months. The low ΔT syndrome was detected with an accuracy of 95.5%, where the DBN had a sensitivity of 0.81, specificity of 0.98 and precision of 0.91. The DBN developed for the school building in Nijmegen had a sensitivity of 0.94, specificity of 0.96, and precision of 0.49, with a fault detection accuracy of 95.7% and a fault diagnosis accuracy of 94.8%.

7.2 Recommendations

Even though the FDD tool can detect the low ΔT syndrome and provide diagnostic results for most of the faults easily, there are certain limitations to its performance and substantial effort is required to make modifications to the tool in terms of adding new projects. To improve the general performance and characteristics of the tool, certain recommendations for future work are proposed as follows:

1. Inclusion of sensor faults in the DBN with the addition of energy balance symptom nodes to provide a complete diagnostic capability.
2. Upgrade of current pomegranate-based DBN to a newer, more user-friendly DBN library to easily configure and update the DBN.

3. Inclusion of additional symptom nodes based on user feedback of thermal comfort to strengthen the diagnostic analysis.
4. Improve the current back-end code base to improve the interaction between different screens and add new projects.

7.3 Publications

As a part of this research, a conference paper has been published at the CLIMA 2022 conference held in Rotterdam, under the topic Digitization. The paper is titled "*Detection of the low ΔT syndrome using machine learning models*".

References

Bruton, K., Raftery, P., Kennedy, B., Keane, M. M., & O'Sullivan, D. T. J. (2014). Review of automated fault detection and diagnostic tools in air handling units. *Energy Efficiency*, *7*(2), 335–351. https://doi.org/10.1007/s12053-013-9238-2

CBS. (2019). *Climate data De Bilt; temperature, precipitation, sunshine 1800-2014*. https://www.cbs.nl/en-gb/figures/detail/80370eng

Chakraborty, D., & Elzarka, H. (2019). Early detection of faults in HVAC systems using an XGBoost model with a dynamic threshold. *Energy and Buildings*, *185*, 326–344. https://doi.org/10.1016/j.enbuild.2018.12.032

Corten, C. J. . (2019). Energy performance optimization of buildings using data mining techniques Energy performance optimization of buildings using data mining. *CLIMA 2019 Congress*. https://doi.org/https://doi.org/10.1051/e3sconf/201911105016

Dai, M., Lu, X., & Xu, P. (2021). Causes of low delta-T syndrome for chilled water systems in buildings. *Journal of Building Engineering*, *33*(February 2020), 101499. https://doi.org/10.1016/j.jobe.2020.101499

Dalpiaz, F., & Brinkkemper, S. (2018). Agile requirements engineering with user stories. *Proceedings - 2018 IEEE 26th International Requirements Engineering Conference, RE 2018*, 506–507. https://doi.org/10.1109/RE.2018.00075

Firrantello, J., Bahnfleth, W., & Kremer, P. (2018). Field measurement and modeling of UVC cooling coil irradiation for heating, ventilating, and air conditioning energy use reduction (RP-1738)—Part 1: Field measurements. *Science and Technology for the Built Environment*, *24*(6), 588–599. https://doi.org/10.1080/23744731.2017.1402662

Francisco J., D., & Marek J., D. (2007). *Canonical Probabilistic Models for Knowledge Engineering* (Vol. 9).

Friedman, H., & Piette, M. A. (2001). Comparative guide to emerging diagnostic tools for large commercial HVAC systems. *Lawrence Berkeley National Laboratory*. https://eta.lbl.gov/publications/comparative-guide-emerging-diagnostic

Gao, D., Wang, S., Shan, K., & Yan, C. (2016). A system-level fault detection and diagnosis method for low delta-T syndrome in the complex HVAC systems. *Applied Energy*, *164*, 1028–1038. https://doi.org/10.1016/j.apenergy.2015.02.025

Gao, D., Wang, S., Sun, Y., & Xiao, F. (2012). Diagnosis of the low temperature difference syndrome in the chilled water system of a super high-rise building: A case study. *Applied Energy*, *98*, 597–606. https://doi.org/10.1016/j.apenergy.2012.03.057

Gao, D., Wang, S., Xiao, F., & Shan, K. (2014). A fault detection and diagnosis method for low delta-T syndrome in a complex air-conditioning system. *Energy Procedia*, *61*, 2514–2517. https://doi.org/10.1016/j.egypro.2014.12.035

Kirsner, W. (1995). Troubleshooting: Chilled Water Distribution Problems at the NASA Johnson Space Center. *Heating, Piping and Air Conditioning*, *February*, 51–59.

Kirsner, W. (1996). The demise of the primary-secondary pumping paradigm for chilled water plant design. *HPAC Heating, Piping, Air Conditioning*, *68*(11), 73–78. https://doi.org/10.1016/s0140-6701(97)85047-3

Lhotka, O., Kyselý, J., & Farda, A. (2018). Climate change scenarios of heat waves in Central Europe and their uncertainties. *Theoretical and Applied Climatology*, *131*(3–4), 1043–1054. https://doi.org/10.1007/s00704-016-2031-3

Marquart, E., & Lange, R. (2017). Monitor energiebesparing gebouwde omgeving. *Rijksdienst Voor Ondernemend Nederland*, *31*(0), 6–7. www.RVO.nl./onderwerpen/duurzaam-ondernemen/gebouwen/

Mills, E. (2011). Building commissioning: A golden opportunity for reducing energy costs and greenhouse gas emissions in the United States. *Energy Efficiency*, *4*(2), 145–173. https://doi.org/10.1007/s12053-011-9116-8

Mo, H., Sun, H., Liu, J., & Wei, S. (2019). Developing window behavior models for residential buildings using XGBoost algorithm. *Energy and Buildings*, *205*, 109564. https://doi.org/10.1016/j.enbuild.2019.109564

Montazeri, A., & Kargar, S. M. (2020). Fault detection and diagnosis in air handling using data-driven methods. *Journal of Building Engineering*, *31*(September 2019), 101388. https://doi.org/10.1016/j.jobe.2020.101388

Pan, B. (2018). Application of XGBoost algorithm in hourly PM2.5 concentration prediction. *IOP Conference Series: Earth and Environmental Science*, *113*(1). https://doi.org/10.1088/1755-1315/113/1/012127

Ruuska, S., Hämäläinen, W., Kajava, S., Mughal, M., & Matilainen, P. (2018). Evaluation of the confusion matrix method in the validation of an automated system for measuring feeding behaviour of cattle. *Behavioural Processes*, *148*(March 2017), 56–62. https://doi.org/10.1016/j.beproc.2018.01.004

RVO, R. voor O. (2021). *Monitor Energiebesparing Gebouwde Omgeving*. https://www.rvo.nl/files/file/2021/12/monitor-energiebesparing-gebouwde-omgeving-2021.pdf

Schein, J., Bushby, S. T., Castro, N. S., & House, J. M. (2006). A rule-based fault detection method for air handling units. *Energy and Buildings*, *38*(12), 1485–1492. https://doi.org/10.1016/j.enbuild.2006.04.014

Schreiber, J. (2018). pomegranate: Fast and flexible probabilistic modeling in python. *Journal of Machine Learning Research*, *18*, 1–6.

Sembian, S. (2019). *A Case Study on Performance Degradation of Cooling Coils in the Netherlands*. MSc Thesis. Eindhoven University of Technology.

Seyedzadeh, S., Rahimian, F. P., Glesk, I., & Roper, M. (2018). Machine learning for estimation of building energy consumption and performance: a review. *Visualization in Engineering*, *6*(1). https://doi.org/10.1186/s40327-018-0064-7

Song, L., & Ph, D. (2011). *Feasibility Study of Developing a Virtual Chilled Water Flow Meter at Air Handling Unit Level*. 1–7.

Taal, A., & Itard, L. (2020). Energy & Buildings P & ID-based automated fault identification for energy performance diagnosis in HVAC systems : 4S3F method , development of DBN models and application to an ATES system. *Energy & Buildings*, *224*, 110289. https://doi.org/10.1016/j.enbuild.2020.110289

Taylor, S. T. (2002). Degrading chilled water plant delta-T: Causes and mitigation. *ASHRAE Transactions*, *108 PART 1*, 641–653.

Walker, S., Khan, W., Katic, K., Maassen, W., & Zeiler, W. (2020). Accuracy of different machine learning algorithms and added-value of predicting aggregated-level energy performance of commercial buildings. *Energy and Buildings*, *209*(February), 109705. https://doi.org/10.1016/j.enbuild.2019.109705

Wang, S., & Jiang, Z. (2004). Valve fault detection and diagnosis based on CMAC neural networks. *Energy and Buildings*, *36*(6), 599–610. https://doi.org/10.1016/j.enbuild.2004.01.037

Yan, C., Yang, X., & Xu, Y. (2018). Mathematical explanation and fault diagnosis of low delta-T syndrome in building chilled water systems. *Buildings*, *8*(7). https://doi.org/10.3390/buildings8070084

Yao, L., Cai, M., Chen, Y., Shen, C., Shi, L., & Guo, Y. (2019). Prediction of antiepileptic drug treatment outcomes of patients with newly diagnosed epilepsy by machine learning. *Epilepsy and Behavior*, *96*, 92–97. https://doi.org/10.1016/j.yebeh.2019.04.006

Zhang, R., & Hong, T. (2017). Modeling of HVAC operational faults in building performance simulation. *Applied Energy*, *202*(April), 178–188. https://doi.org/10.1016/j.apenergy.2017.05.153

Zhao, Y., Li, T., Zhang, X., & Zhang, C. (2019). Artificial intelligence-based fault detection and diagnosis methods for building energy systems: Advantages, challenges and the future. *Renewable and Sustainable Energy Reviews*, *109*(February), 85–101. https://doi.org/10.1016/j.rser.2019.04.021

Zhou, Z.-H., & Liu, S. (2021). *Machine Learning* (1st ed.). Springer Singapore. https://doi.org/https://doi.org/10.1007/978-981-15-1967-3

Appendix

A1. Validation of simulation model of office building in Breda
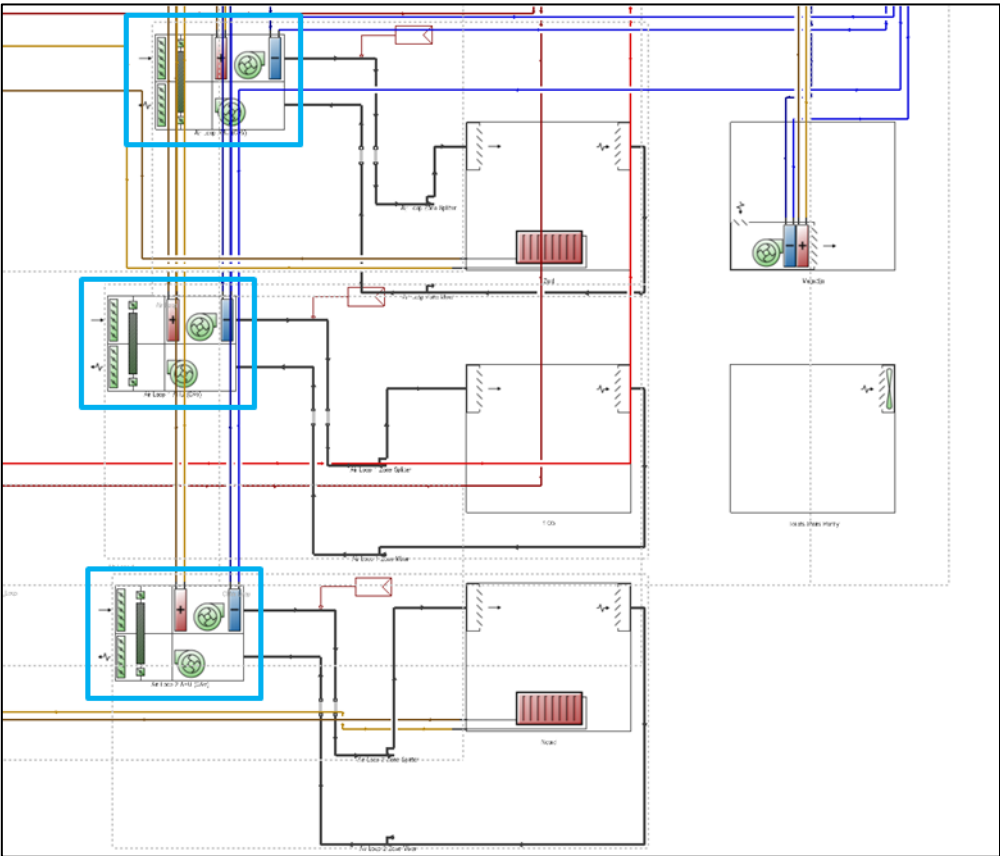
Figure 61: HVAC layout of office building in Breda designed in DesignBuilder
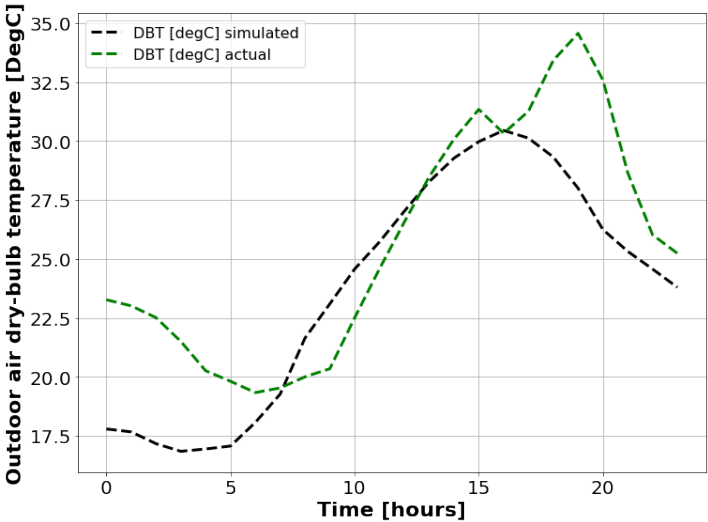
Figure 62: Comparison of simulated and actual dry-bulb outdoor air temperature
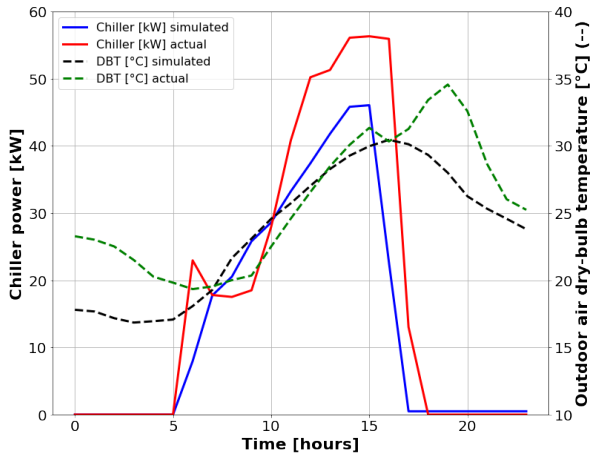
Figure 63: Comparison of simulated and actual chiller power for high outdoor air temperature case (MBE = 8.9 kW)
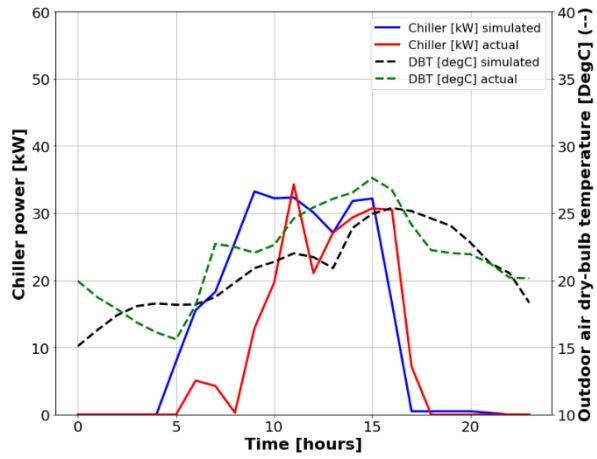
Figure 64: Comparison of simulated and actual chiller power for low outdoor air temperature case (MBE = 5.1 kW)
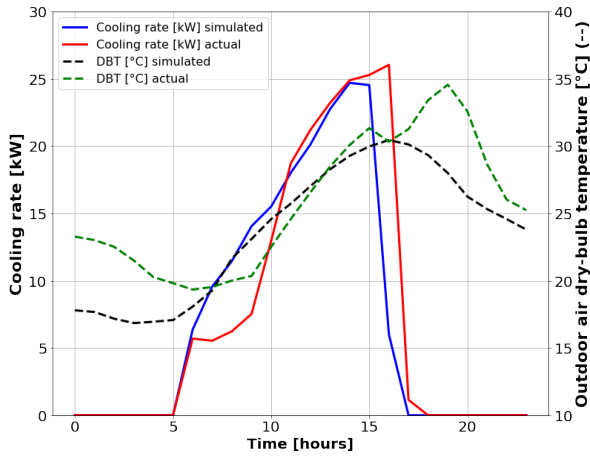
Figure 65: Comparison of simulated and actual cooling rate for high outdoor air temperature case (MBE = 1.2 kW)
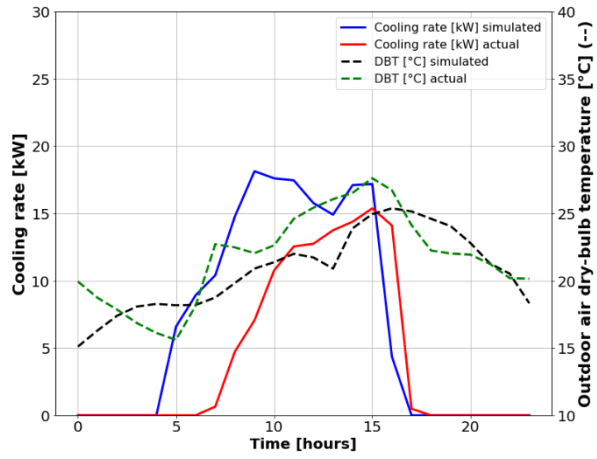
Figure 66: Comparison of simulated and actual cooling rate for low outdoor air temperature case (MBE = 3.1 kW)
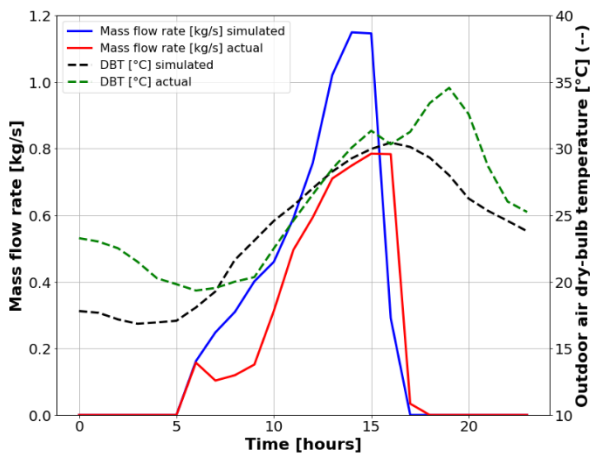
Figure 67: Comparison of simulated and actual MFR for high outdoor air temperature case (MBE = 0.13 kg/s)
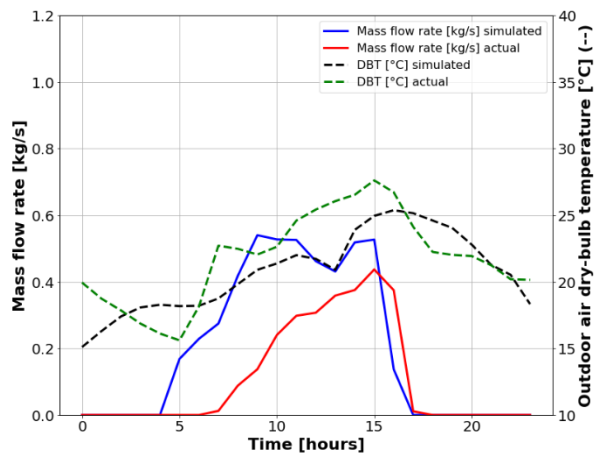
Figure 68: Comparison of simulated and actual MFR for high outdoor air temperature case (MBE = 0.16 kg/s)
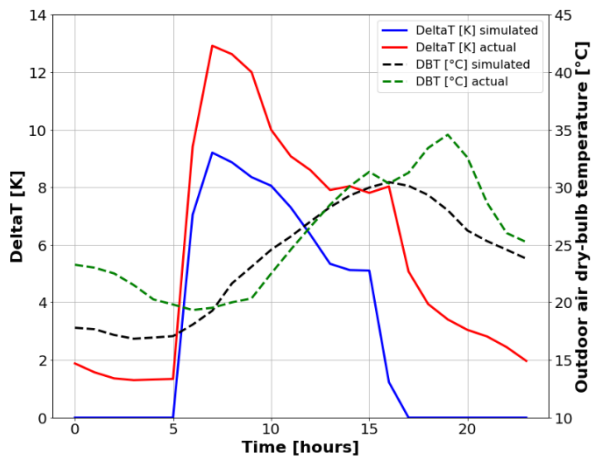
Figure 69: Comparison of simulated and actual ΔT for high outdoor air temperature case (MBE = 3.3 K)
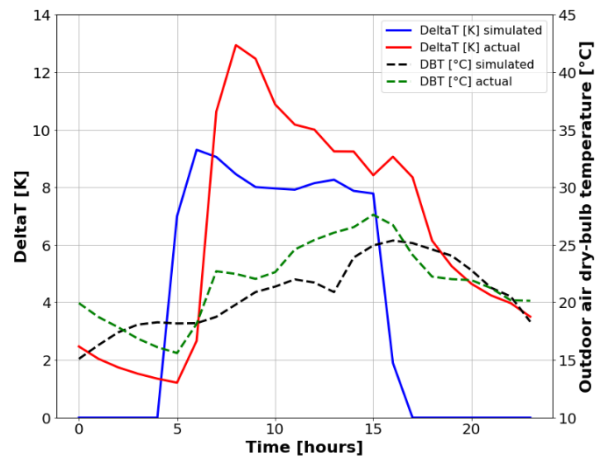
Figure 70: Comparison of simulated and actual ΔT for low outdoor air temperature case (MBE = 3 K)

A2. Fault impact analysis of 5-zone office building
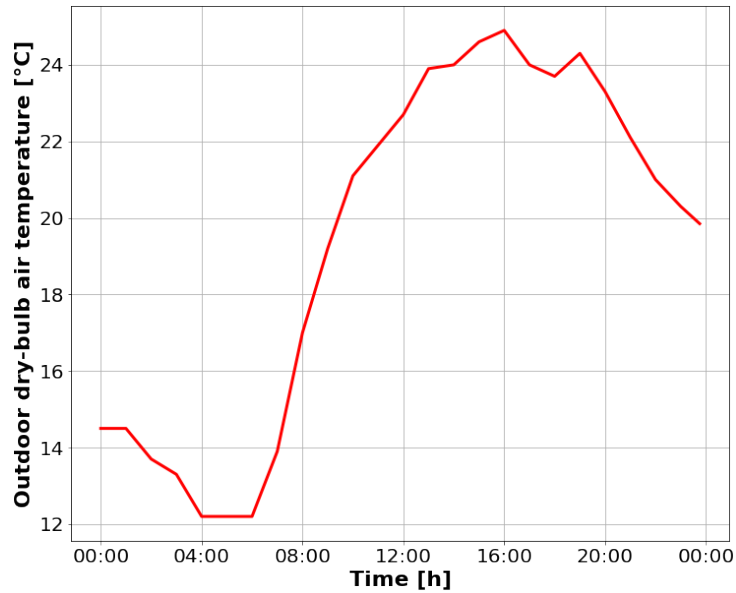
Figure 71: Temperature profile of outdoor air for a reference day used in the simulation of the 5-zone model
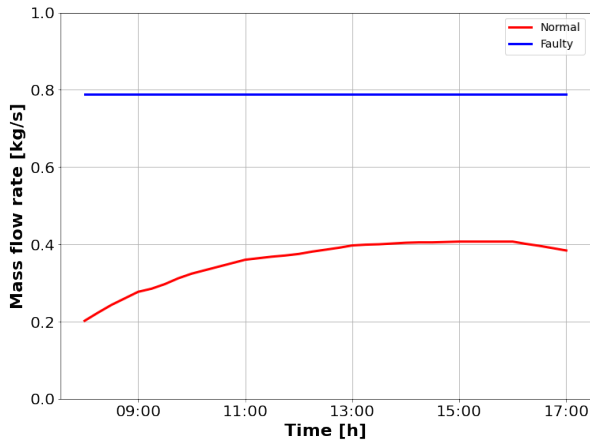
Figure 72: Comparison of normal and faulty mass flow rate for 75% stuck valve fault of 5-zone building
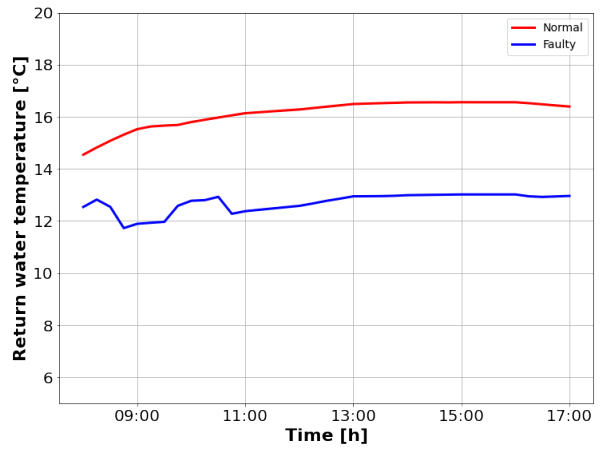
Figure 73: Comparison of normal and faulty return water temperature for 75% stuck valve fault of 5-zone building
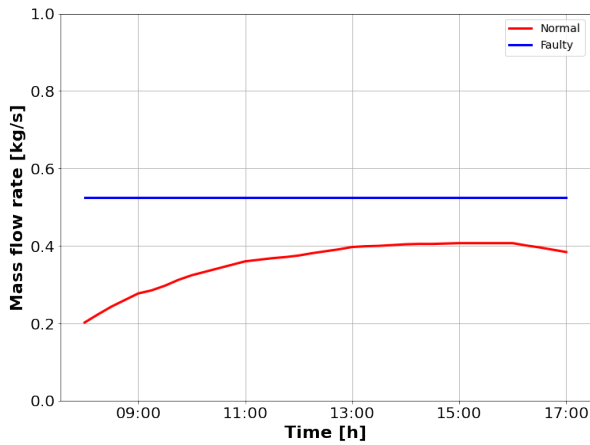
Figure 74: Comparison of normal and faulty mass flow rate for 50% stuck valve fault of 5-zone building
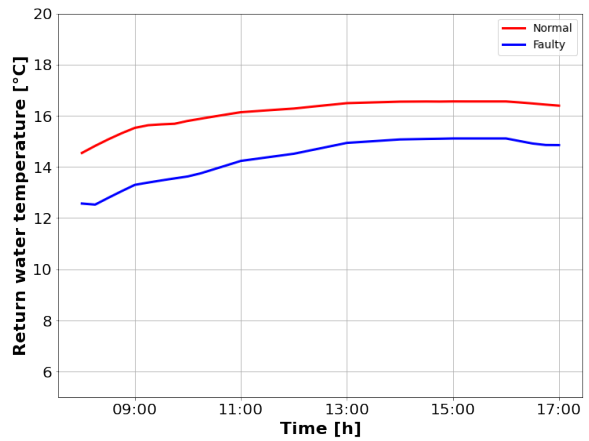
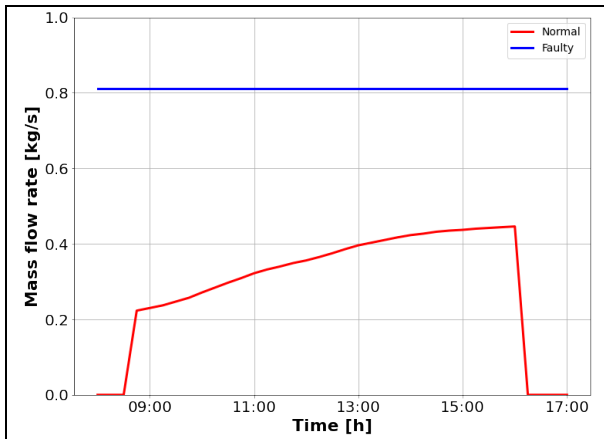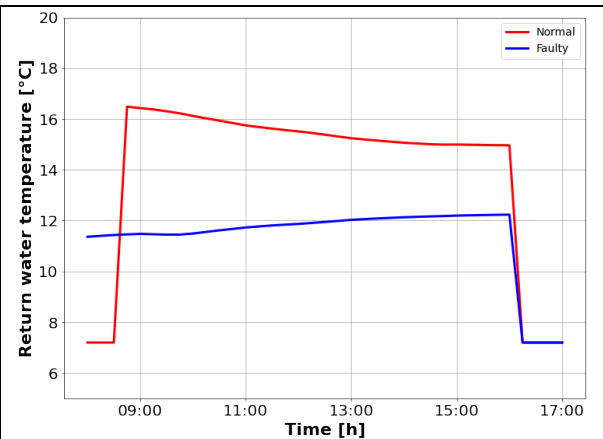Figure 75: Comparison of normal and faulty return water temperature for 50% stuck valve fault of 5-zone building
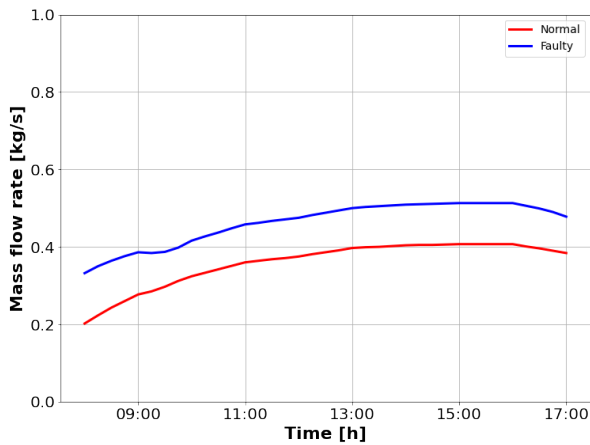
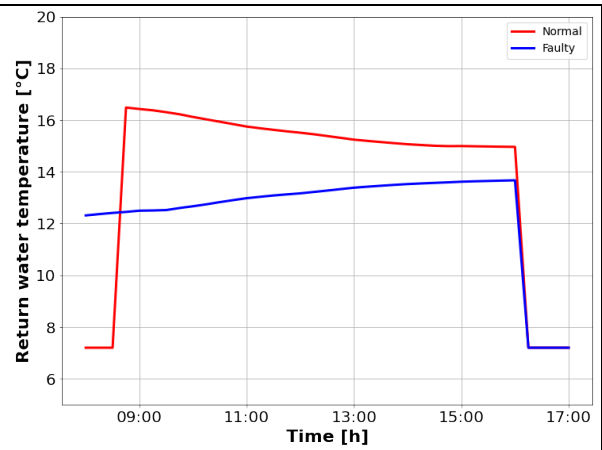Figure 76: Comparison of normal and faulty mass flow rate for reduced SAT by 1K fault of 5-zone building
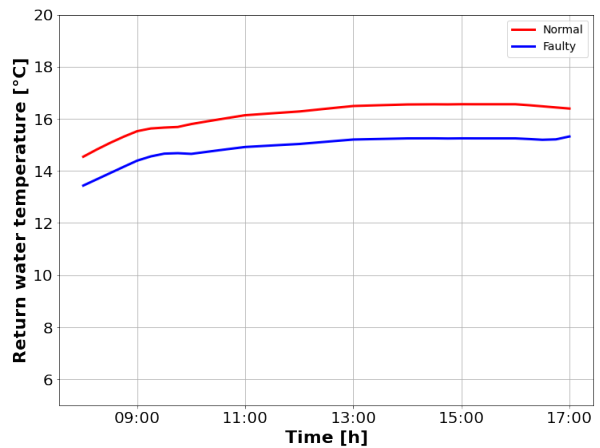
Figure 77: Comparison of normal and faulty return water temperature for reduced SAT by 1K fault of 5-zone building
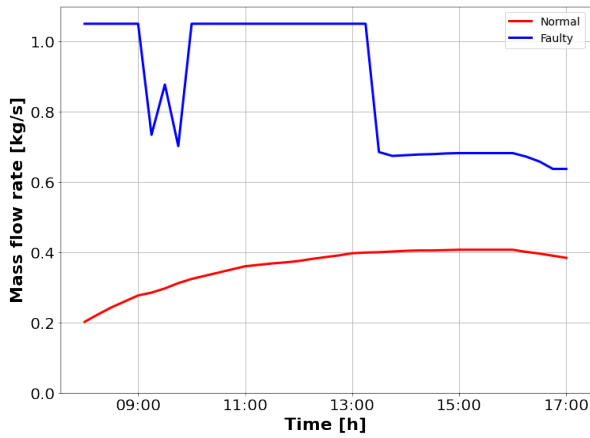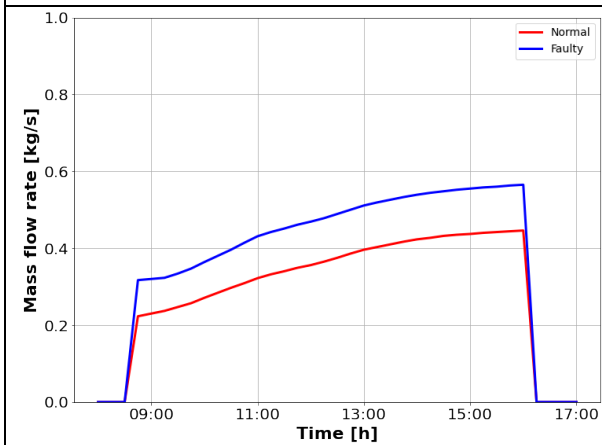
Figure 78: Comparison of normal and faulty mass flow rate for reduced SAT by 2K fault of 5-zone building
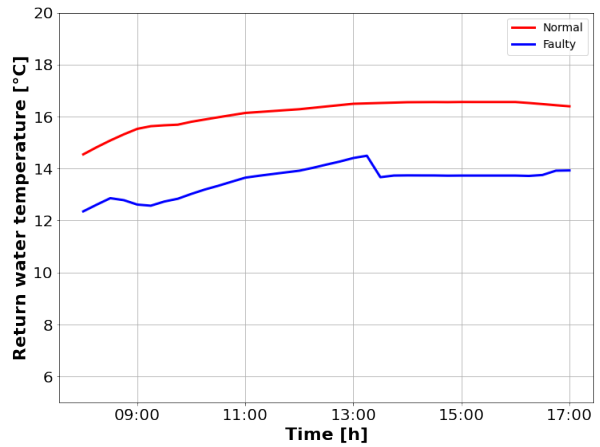
Figure 79: Comparison of normal and faulty return water temperature for reduced SAT by 2K fault of 5-zone building
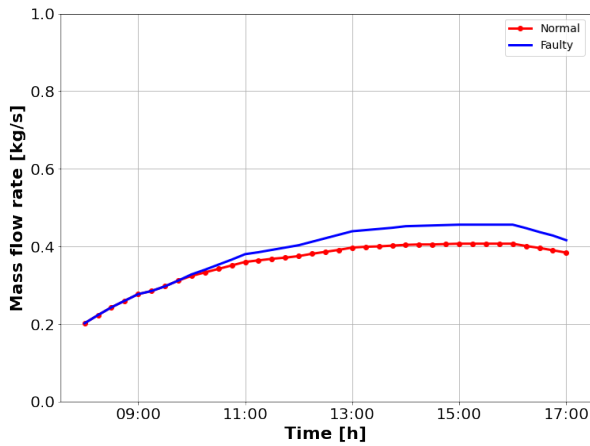
Figure 80: Comparison of normal and faulty mass flow rate for 30% fouling fault of 5-zone building
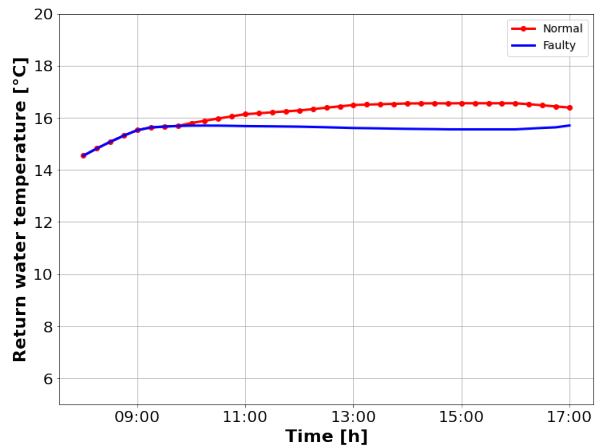
Figure 81: Comparison of normal and faulty return water temperature for 30% fouling fault of 5-zone building
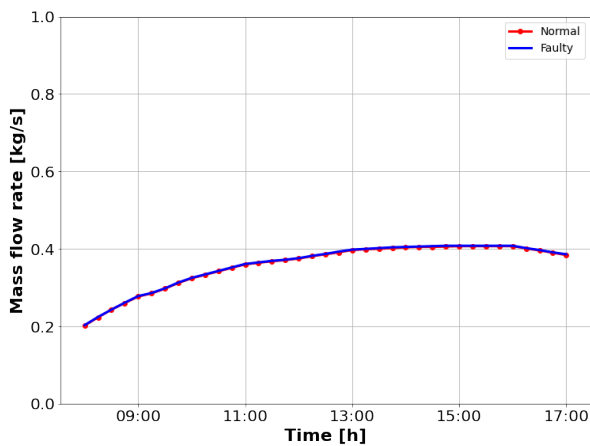
Figure 82: Comparison of normal and faulty mass flow rate for higher airflow fault of 5-zone building
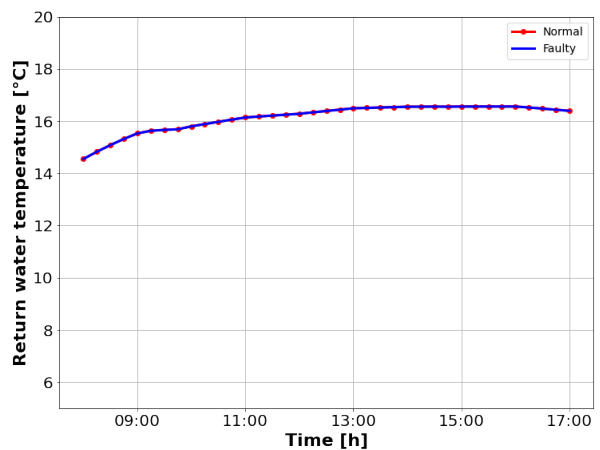
Figure 83: Comparison of normal and faulty return water temperature for higher airflow fault of 5-zone building
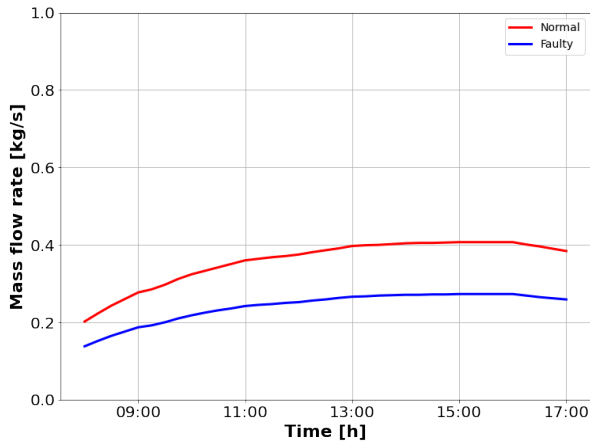
Figure 84: Comparison of normal and faulty mass flow rate for lower airflow fault of 5-zone building
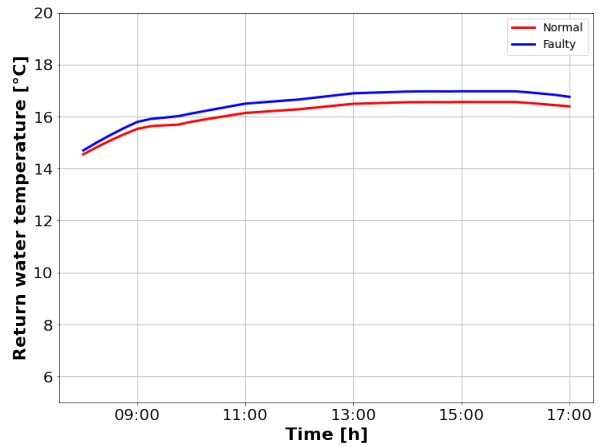
Figure 85: Comparison of normal and faulty return water temperature for lower airflow fault of 5-zone building
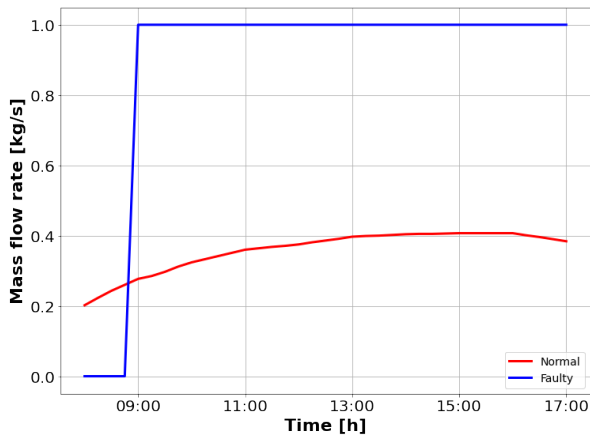
Figure 86: Comparison of normal and faulty mass flow rate for increased SWT fault of 5-zone building
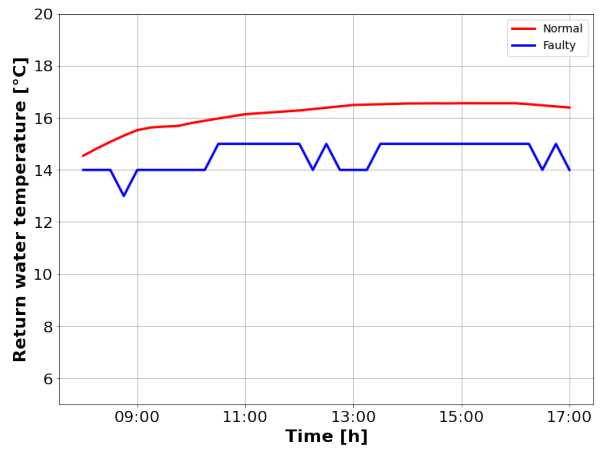
Figure 87: Comparison of normal and faulty return water temperature for increased SAT fault of 5-zone building
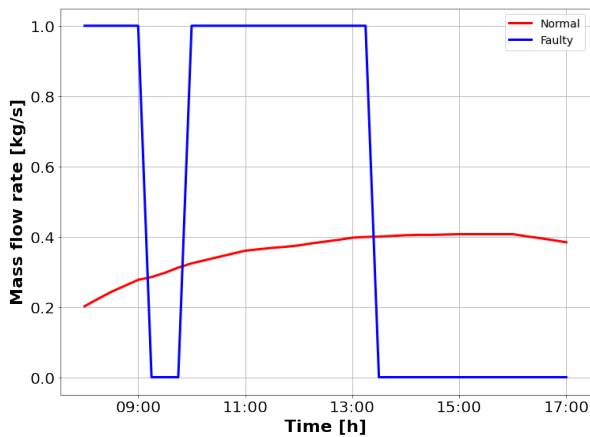
Figure 88: Comparison of normal and faulty mass flow rate for positive sensor offset fault of 5-zone building
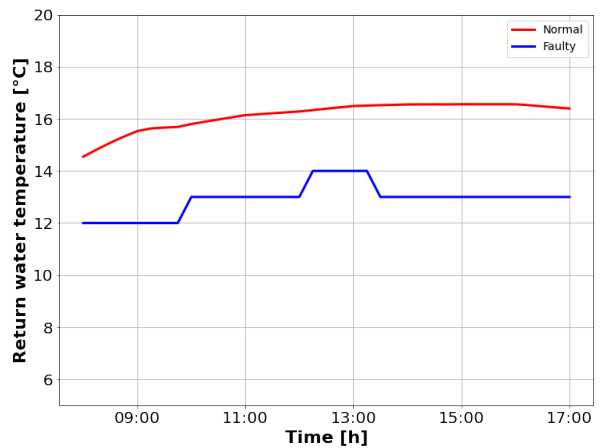
Figure 89: Comparison of normal and faulty return water temperature for positive sensor offset fault of 5-zone building

A3. Fault characteristic analysis of office building in Breda
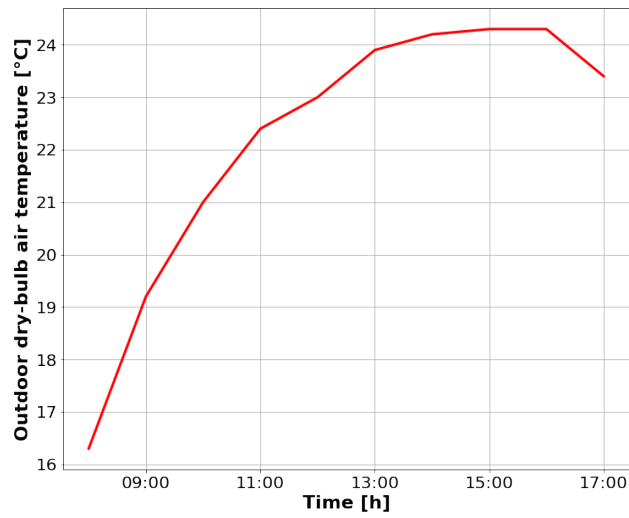
Figure 90: Temperature profile of outdoor dry-bulb air for simulation of office building in Breda during working hours

Figure 91: Comparison of normal and faulty mass flow rate for 75% stuck valve fault at simulation of office building in Breda
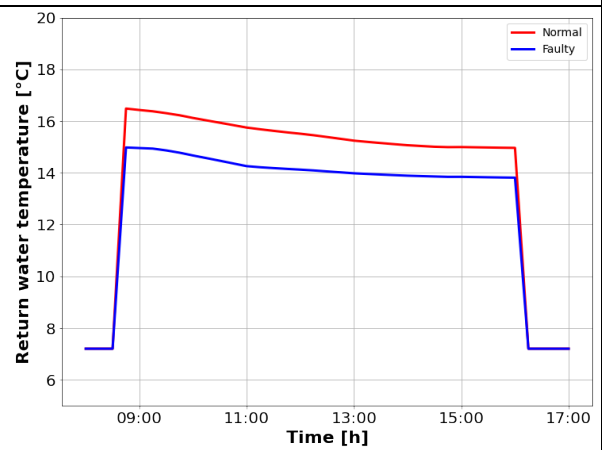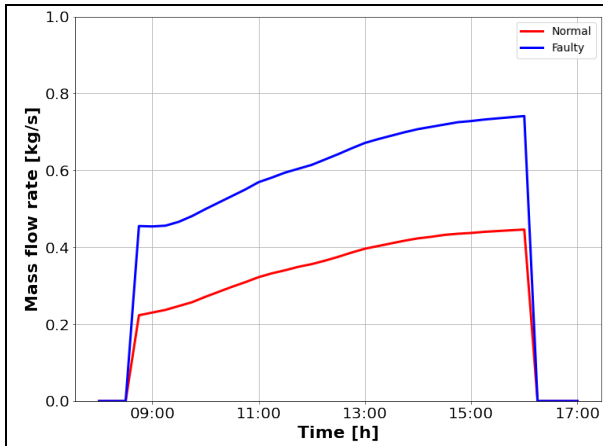

Figure 92: Comparison of normal and faulty return water temperature for 75% stuck valve fault at simulation of office building in Breda
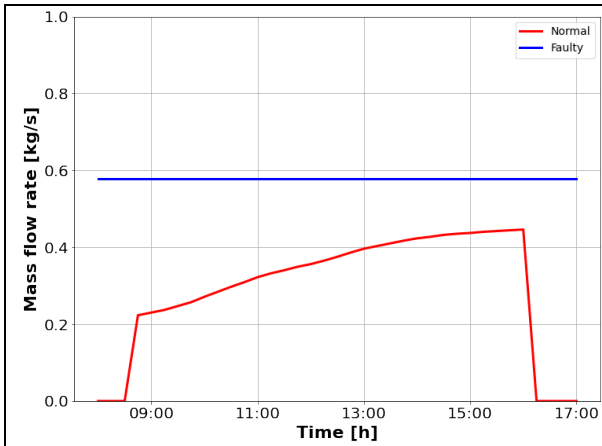
Figure 93: Comparison of normal and faulty mass flow rate for 50% stuck valve fault at simulation of office building in Breda

Figure 94: Comparison of normal and faulty return water temperature for 50% stuck valve fault at simulation of office building in Breda

Figure 95: Comparison of normal and faulty mass flow rate for reduced SAT by 1K fault at simulation of office building in Breda
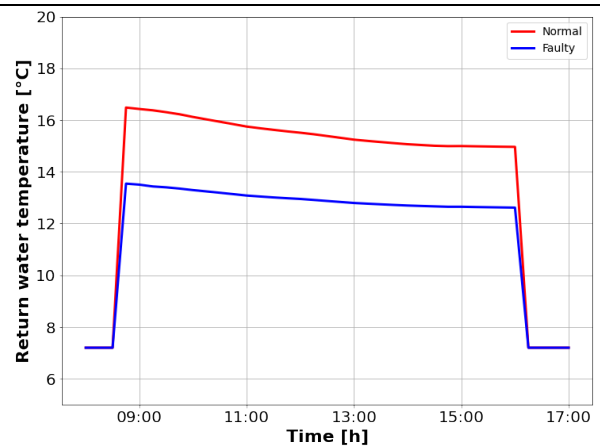

Figure 96: Comparison of normal and faulty return water temperature reduced SAT by 1K fault at simulation of office building in Breda

Figure 97: Comparison of normal and faulty mass flow rate for reduced SAT by 2K fault at simulation of office building in Breda

Figure 98: Comparison of normal and faulty return water temperature reduced SAT by 2K fault at simulation of office building in Breda

A4. Noisy-max algorithm development

To develop a graphical probabilistic model such as a Bayesian network, it is necessary to provide CPTs which consist of discrete joint probability distributions between the different states of the parent nodes and the child node. With an increase in parent nodes however, the size of the CPT increases exponentially. This is where canonical models are used which reduce the complexity of elicitation of numerical probabilities (Francisco J. & Marek J., 2007). There are three types of canonical models – deterministic, indeterministic and simple canonical models. Deterministic models are obtained from logical and algebraic functions and do not require any numerical parameters. Indeterministic models are based on the assumption of independence of causal influence (ICI) and are further sub-divided into noisy and leaky models. The conversion of a noisy MAX probability distribution to a CPT probability distribution follows a defined procedure. For the FDD tool, the leaky causal MAX algorithm is used since it is the same conversion algorithm used in GeNie. Figure 99 shows the structure of a noisy MAX canonical model. Here X is the original parent node, Z is an auxiliary variable and Y is the child node such that Y is a deterministic function of the Zs and each Z is probabilistically dependant on X. There can be in total n such parent nodes for a particular child node.
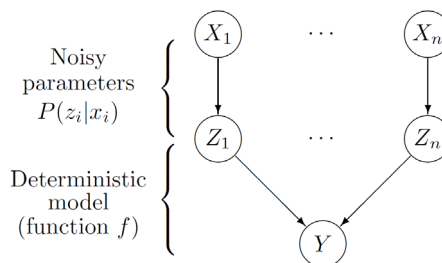

Figure 99: Internal structure of a noisy MAX canonical model

The conversion algorithm used for noisy MAX to general CPTs is based on the following set of equations

63

$$P(Y \leq y|\text{X}) = \left(\sum_{z_L \leq y} c_{z_L}^L\right) \cdot \prod_i \left(\sum_{z_i \leq y} c_{z_i}^{x_i}\right)$$

(3)

$$P(Y \leq y|\text{X}) = C_y^L \cdot \prod_i C_y^{x_i}$$

(4)

$$P(y|x) = \begin{cases} P(Y \leq y|\text{X}) - P(Y \leq y - 1|\text{X}) & for \; y \neq y_{min} \\ P(Y \leq y|\text{X}) & for \; y = y_{min} \end{cases}$$

(5)

Here, $P(y|x)$ is the final generated CPT for each state x of parent node X_i in the parent set X, and y in the child node Y, where the set $\text{X} = \{X_1, X_2, \dots, X_n\}$. Here, z_i is the individual auxiliary state of Z_i, C_y^L is the sum of all the leak probability distribution values in the auxiliary distribution Z. A leak probability is the probability of the state y in node Y where the x's of X are in their neutral states, i.e., for example if a stuck valve fault node (X) has a fault-free state, then the probability of the cooling coil valve position symptom node (Y) with the fault-free state (no residuals) is the leak probability. Therefore, the final probability distribution corresponding to each state of the child node and the specific combination of states of the parent nodes $P(y|x)$ is calculated using Equation (5). The different values of $P(y|x)$ for all the different states constitute the complete CPT for a particular child-parent acyclic graph

A5. FDD algorithm development for school building in Nijmegen
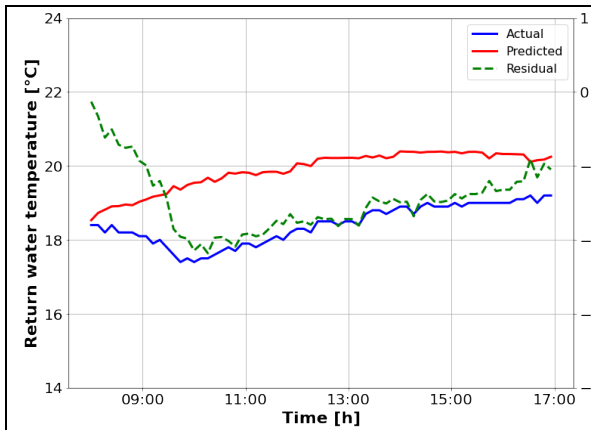

Figure 100: Comparison of predicted and actual RWT for reduced SAT fault by 2K conducted at school building in Nijmegen on 08/08/2020
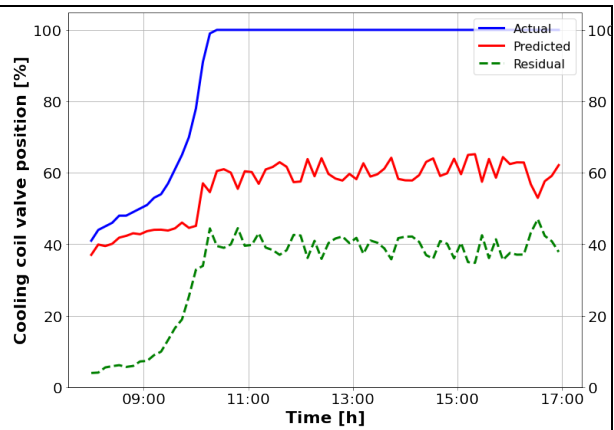

Figure 101: Comparison of predicted and actual CCVP for reduced SAT fault by 2K conducted at school building in Nijmegen on 08/08/2020
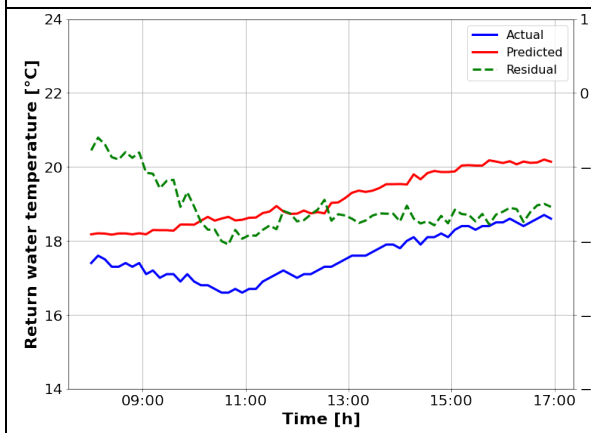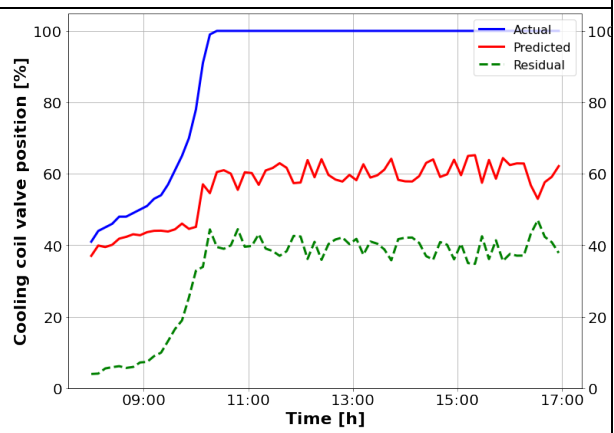

Figure 102: Comparison of predicted and actual RWT for reduced SAT fault by 2K conducted at school building in Nijmegen on 09/08/2020

Figure 103: Comparison of predicted and actual CCVP for reduced SAT fault by 2K conducted at school building in Nijmegen on 09/08/2020
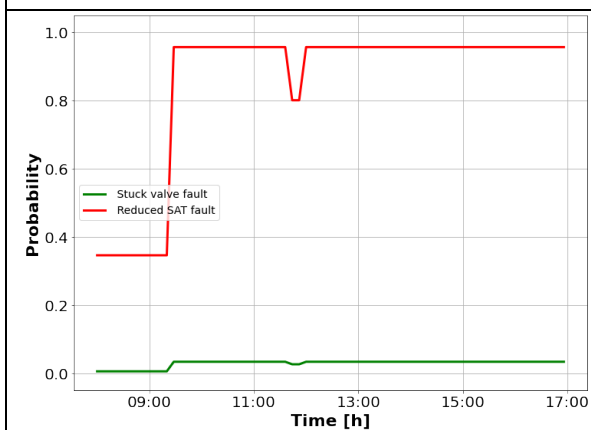

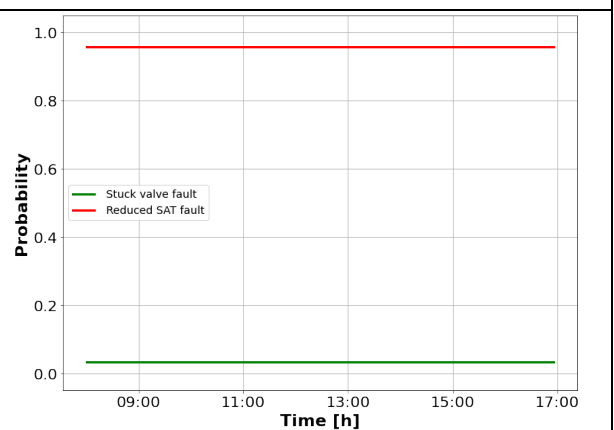Figure 104: Posterior probability distribution of all faults on 08/08/2020

Figure 105: Posterior probability distribution of all faults on 09/08/2020
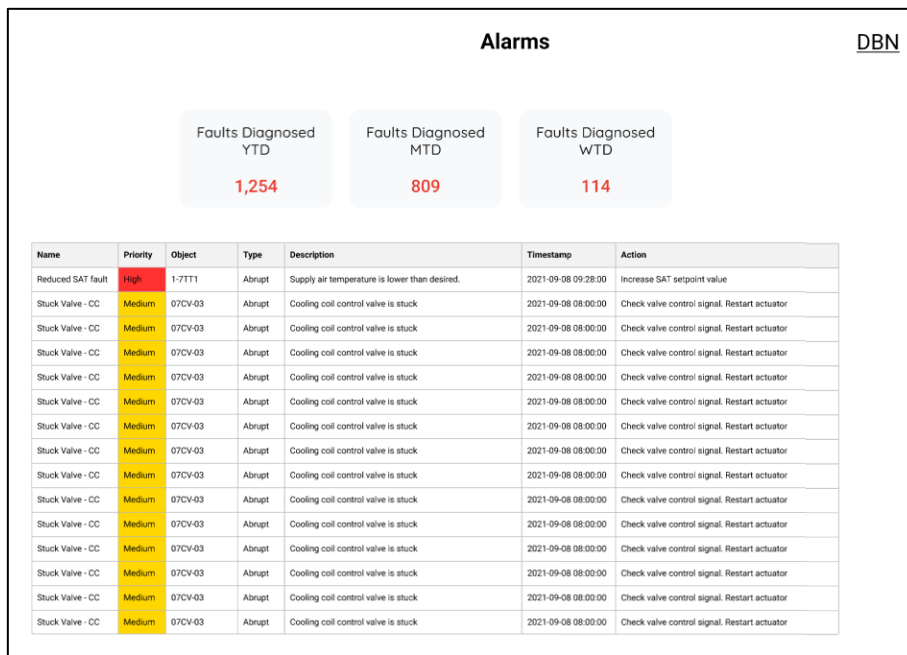
A6. Figma screens during prototype development

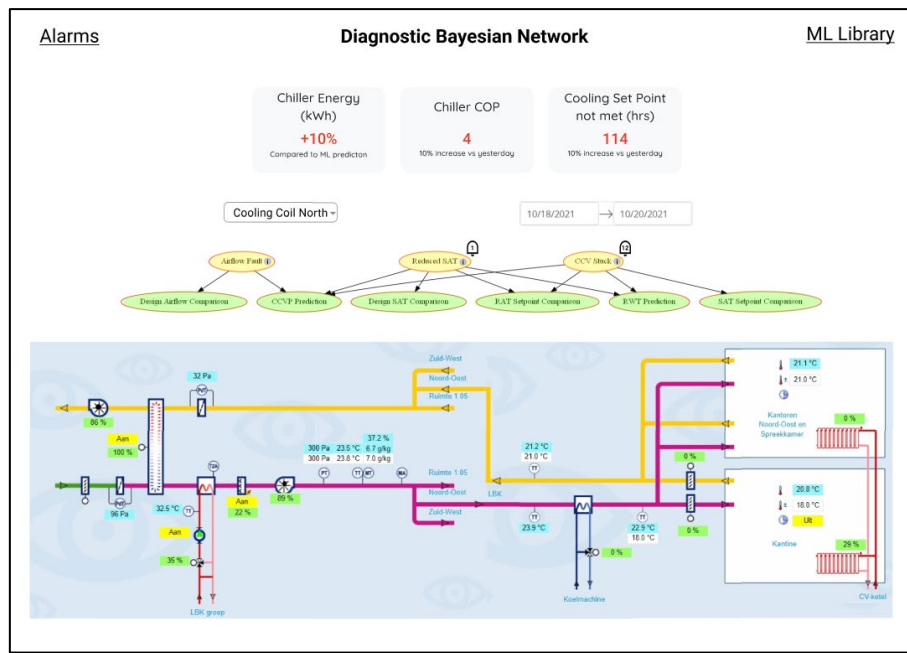Figure 106: Figma screen of alarms page
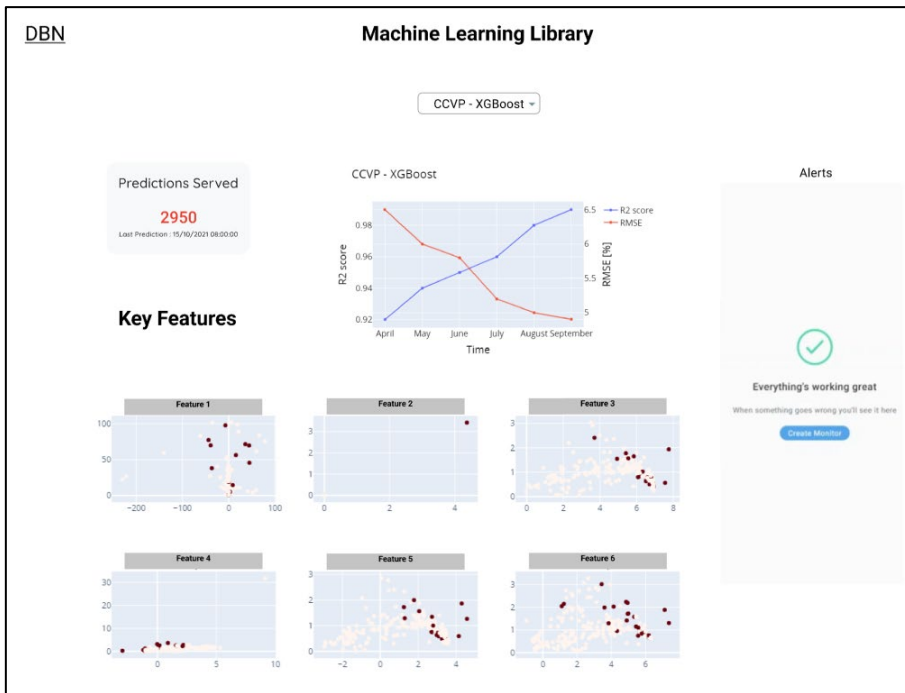

Figure 107: Figma screen of DBN page
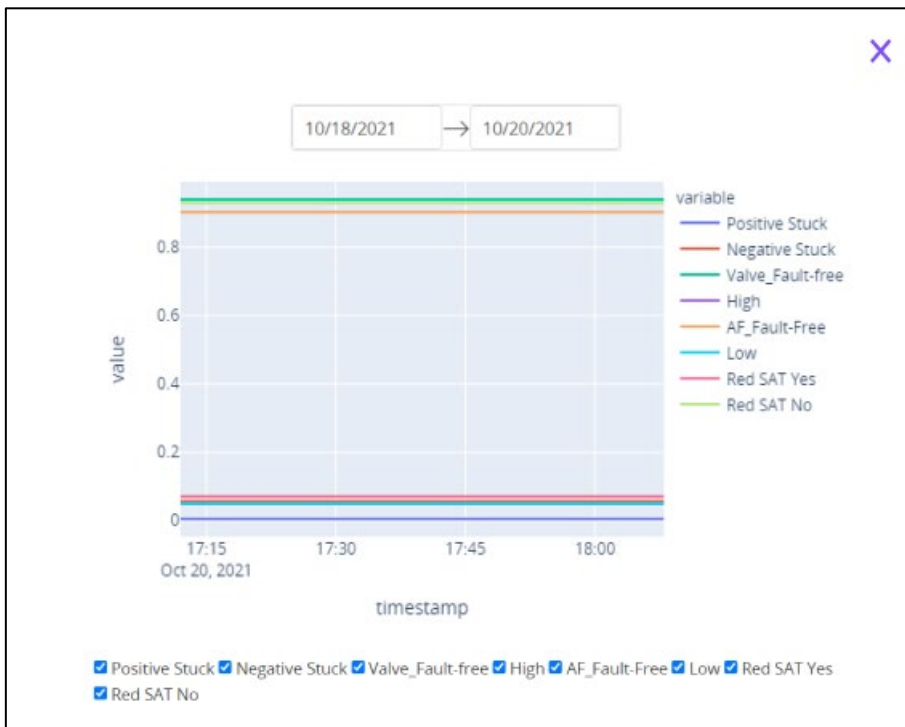
Figure 108: Figma screen of ML page
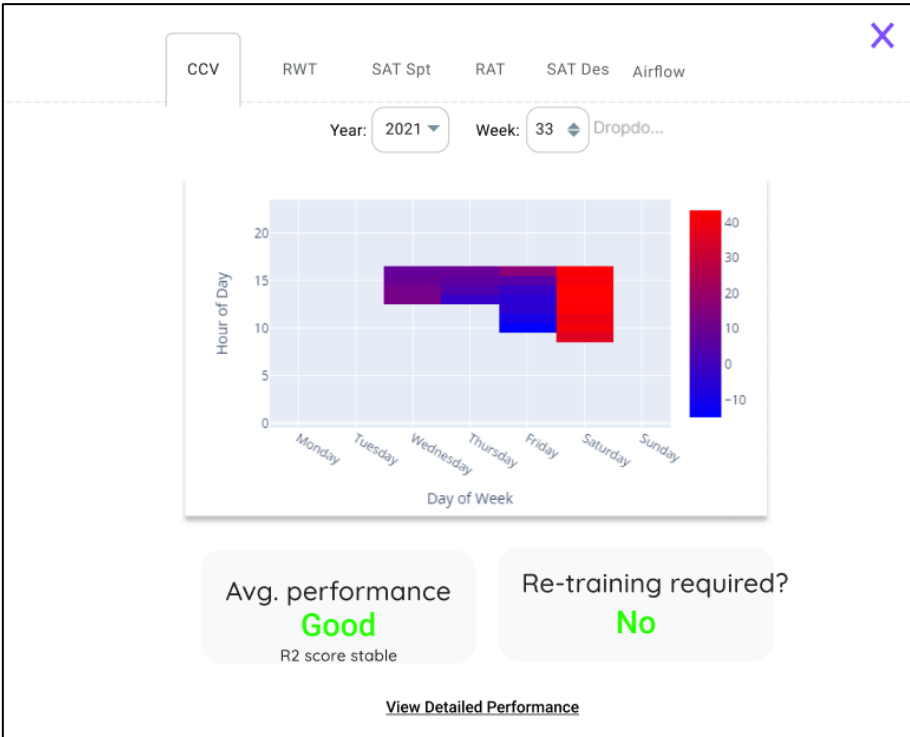

Figure 109: Figma screen of fault probability modal

Figure 110: Figma screen of symptom node residual modal

A7. Validation of FDD tool for office building in Breda – other fault cases

Figure 111: Screenshot of alarms page for validation of reduced SAT by 2K fault on 03/06/2022
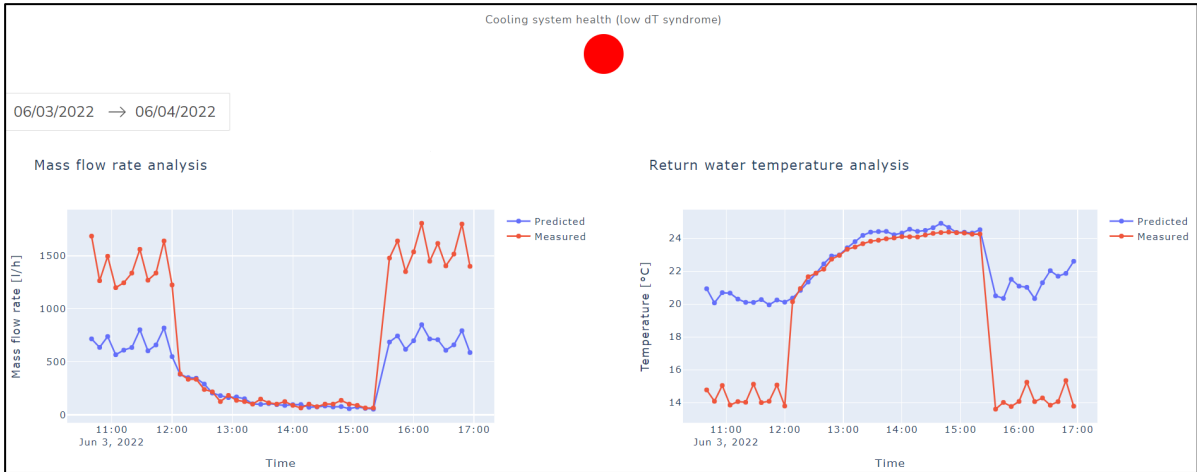
Figure 112: Screenshot of low ΔT syndrome analysis page for validation of reduced SAT by 2K fault on 03/06/2022
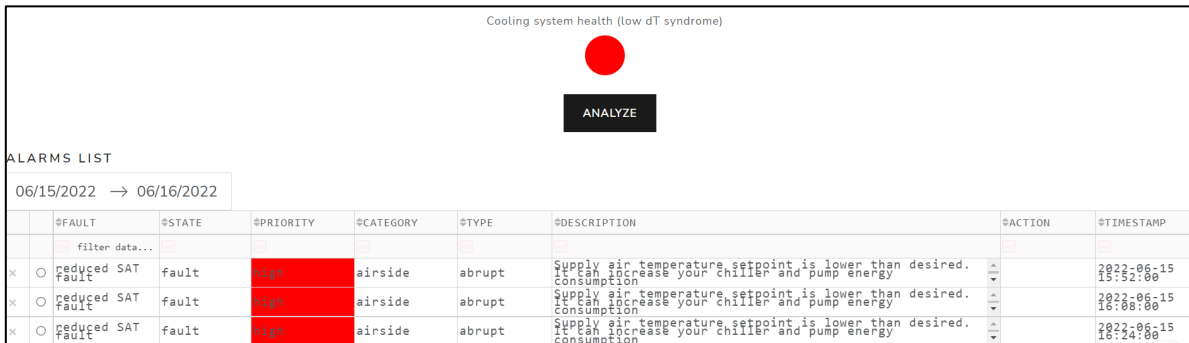
Figure 113: Screenshot of alarms page for validation of reduced SAT by 1K fault on 15/06/2022
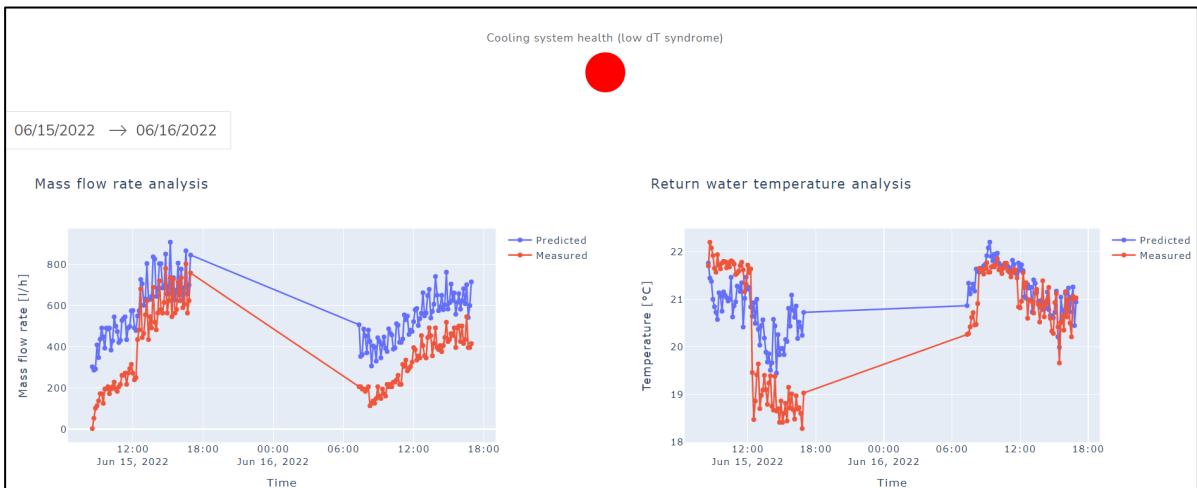
Figure 114: Screenshot of low ΔT syndrome analysis page for validation of reduced SAT by 1K fault on 15/06/2022

A8. Validation of FDD tool for school building in Nijmegen

Figure 115: Screenshot of alarms page for school building in Nijmegen during the reduced SAT by 2K fault on 08/08/2020

Figure 116: Screenshot of alarms page for school building in Nijmegen during the reduced SAT by 2K fault on 09/08/2020

Figure 117: Screenshot of low ΔT syndrome analysis page for school building in Nijmegen during the reduced SAT by 2K fault on 08/08/2020 and 09/08/2020
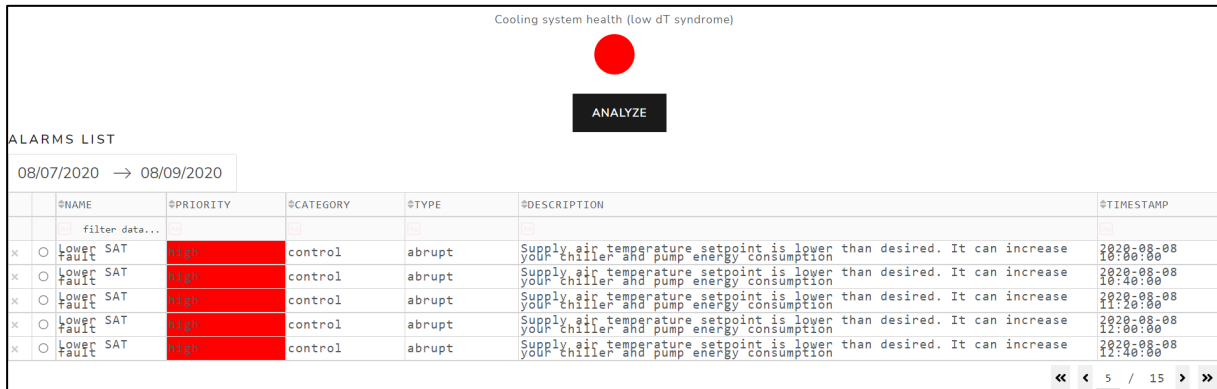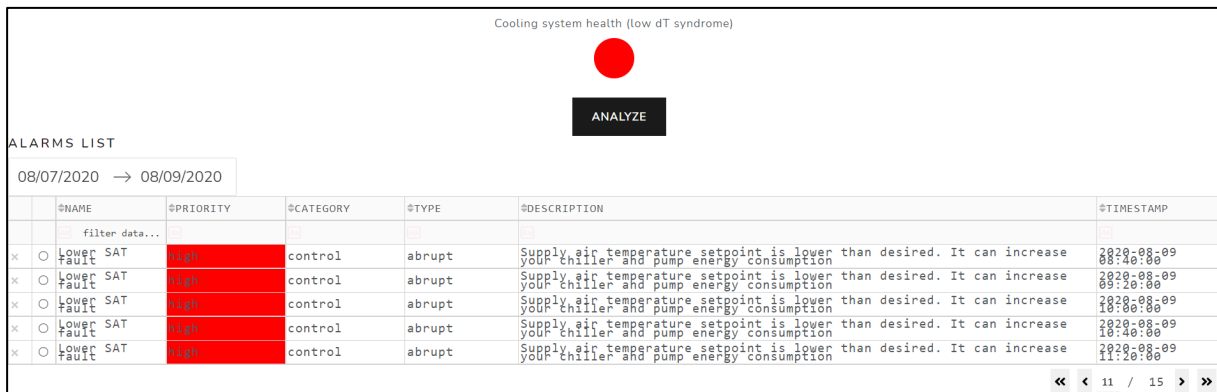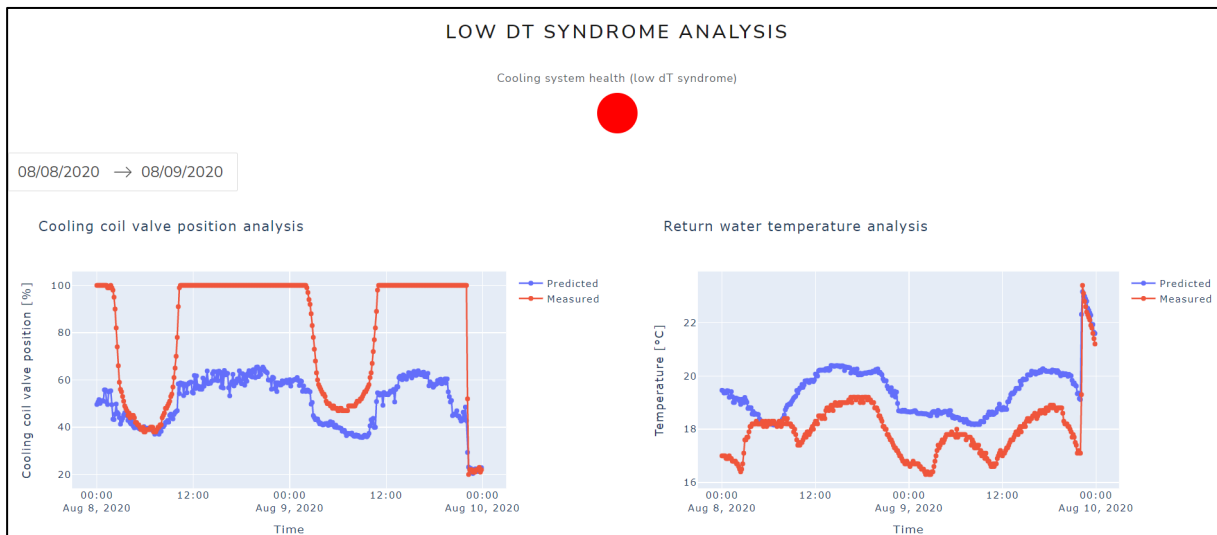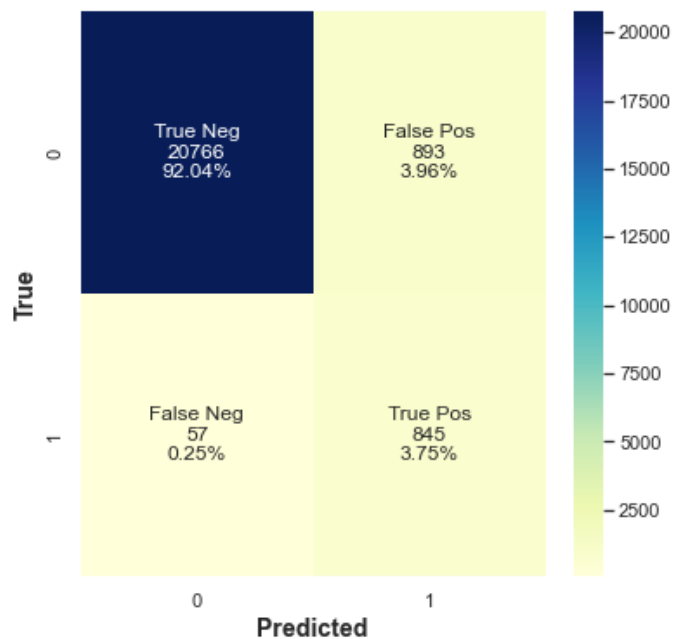
Figure 118: Confusion matrix for DBN of AHU-1 of school building in Nijmegen

The DBN developed for the school building in Nijmegen was able to accurately detect the low ΔT syndrome for 95.8% of the total cases. The DBN was found to have a sensitivity of 0.94, specificity of 0.96 and a precision of 0.49. The overall fault labelling accuracy (diagnosis accuracy) of the tool was calculated to be 94.8%. The precision of the model is not very high, and this is because during the reduced SAT faults conducted at the location collided with very warm days during the summer of 2020, where the outside air temperatures were higher than 30 °C. This led to certain unusual circumstances, where some symptom nodes did not indicate a fault since the

A9. Business plan proposed for the developed FDD tool
Introduction

The Smart HVAC tool is the latest HVAC monitoring tool with capabilities of fault detection and diagnosis of commonly occurring HVAC related faults and auto-correction of control and setpoint related faults. The tool also has the additional function of being able to detect the low ΔT syndrome in cooling coils, which is a decades long problem in the HVAC industry. The predictive maintenance strategy employed by the system allows the user to identify severe faults in the system beforehand.

Situation
The number of commercial buildings in the world are ever increasing to meet the demand of a growing population and workforce. With an increasing average global temperature, the cooling demand in buildings is at its highest and is expected to grow in the coming years. To comply with this demand, HVAC systems are increasingly used in buildings to comply with the comfort requirement of the occupants. With an increased use of HVAC systems, the possibilities of faults occurring in the system are much higher, leading to higher energy consumption of the system and degradation of occupancy comfort. The use of an FDD tool with re-commissioning of systems in the form of predictive maintenance can lead to energy savings of 10-20%.

Problem

Currently, many buildings in the Netherlands do not have a sophisticated FDD tool integrated with the building management system (BMS). The BMS module can detect some very minor faults in the system and provide alarms respectively, but this is not the case for gradual faults or control faults which can lead to excessive energy consumption of the HVAC components. Faults related to the coils like performance degradation and leakage need appropriate methods for detection and diagnosis. Such faults can only be noticed by technicians during regular maintenance.

Solution

One promising solution for such kinds of faults is the use of machine learning, data analytics and pattern recognition methods. Our product, which is a smart fault detection and diagnosis tool, consists of multiple modules based on state-of-the-art artificial intelligence tools. The Smart HVAC tool can continuously monitor the performance of different HVAC components and identity faulty instances due to performance degradation using advanced data analytics. The control and sensor setpoint faults which can occur sporadically, will be detected using machine learning models and will be auto-corrected to operate at optimal operating conditions. The tool can also prevent certain severe faults from occurring by notifying the user of early degradation of HVAC systems.

Unique value proposition and business model

The Smart HVAC ΔT tool is a multi-functional fault detection and diagnosis tool with capability of auto-correction of control and setpoint faults. This minimizes the requirement of human intervention for repairs, reducing labour and energy costs. Since the product offers purely digital and smart solutions, the operational costs to the customer are much lower compared to conventional manual maintenance costs required per year. The product will be sold to the customer in the form of a monthly subscription which includes customer service support and technical support with the tool. Additional maintenance for serious faults will be handled by Kropman Installatietechniek.

Product

The product is an application which is integrated into the BMS platform of a building and connected with the sensor and actuators of all HVAC components. The tool continuously monitors the data streams from all components and analyses the performance of sensors and actuators. Detected faults are displayed in a user-friendly interface with the proper diagnosis of the faults, indication of fault severity, recommended actions and corrective actions taken. The system is also accessible to technical professionals to get a more detailed diagnosis report of faults. A prototype mock-up of the application interface is shown below.
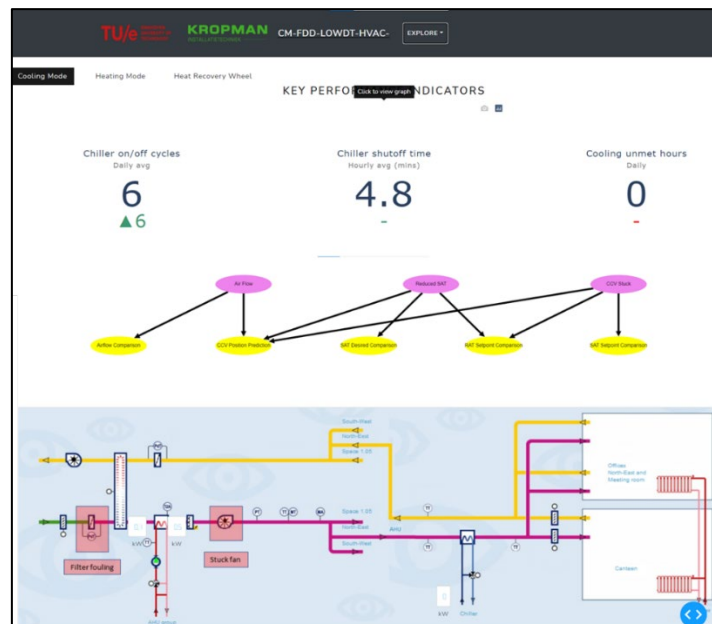
Figure 119: DBN analysis page of the FDD tool

The product can be sold in two forms:

1. BMS + FDD which is aimed for new buildings and re-commissioning buildings where the BMS platform of Kropman Installatietechniek will be used.
2. FDD only system which is aimed for existing buildings having a BMS system but no FDD capabilities.

Market

Market size

Currently there are around 470,000 commercial buildings in the Netherlands, with many of them constantly being renovated and HVAC equipment being recommissioned. Furthermore, there are around 400-500 new buildings built every year. Initially, the product will be targeted to around 10 new buildings in the first year of market introduction with a further expansion to 25 buildings in the second year. With more market penetration in new buildings, the product can be targeted to recommissioning buildings as well.

Target customers

Customers that are most likely to buy the product include property owners of hotels, schools, universities, hospitals, offices, factories and labs. The product would first be sold to existing customers of Kropman Installatietechniek who are using the Kropman BMS system. Since Kropman has already established a strong clientele in the Netherlands, it is relatively easier to sell the product to customers. The Smart HVAC ΔT tool, which is a module of the Kropman BMS system, would be sold at a lower price compared to the annual maintenance contract provided to the customer. The module will then be extended to other customers in the Netherlands, promoting the cheaper and more efficient BMS alternative of Kropman Installatietechniek.

Competitors

There are a few basic diagnostic features available in some BMS and building automation systems
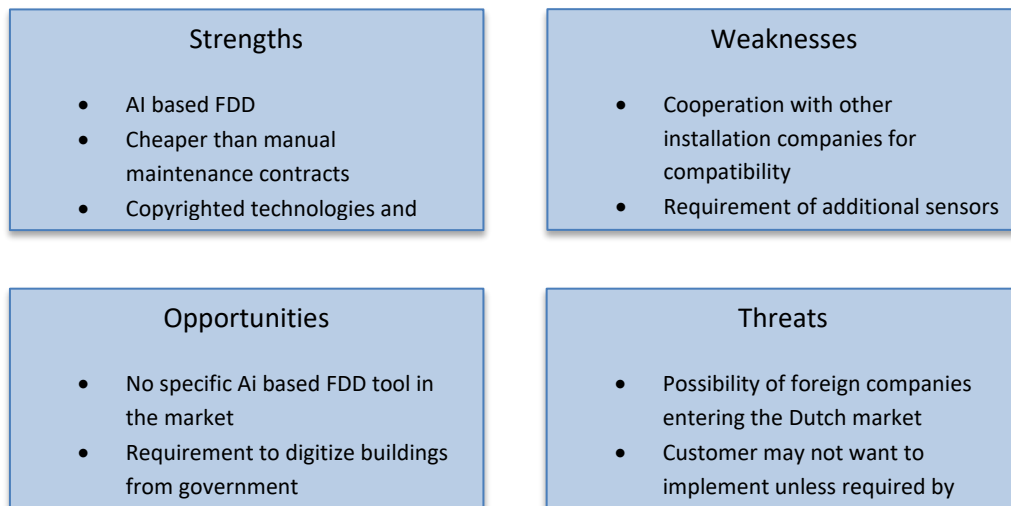
offered currently but they are mostly oriented towards the US HVAC configuration and climate or are detect only very simple and basic faults. The comparison is shown below in Table 11.

Table 11: Market survey of competitors

Tool	Company	Cons
OpenBlue	Johnson Controls	- US only - Not developed for EU - Rules based
Priva Blue	Priva	Basic rules-based method
HVAC-Cx	NIST	- US based - Offline basis - Minimal flexibility with BMS - Rules based
Brainbox AI	Brainbox AI	- Australia and Canada only - Not developed for EU
Clockworks Analytics	Clockworks Analytics	US based

From a customer standpoint, the FDD systems of OpenBlue, HVAC-Cx and Brainbox are not very useful to the Dutch buildings since they are built for a different region. Brainbox AI and Clockworks Analytics are not yet available in the EU and therefore not available to any of the Dutch customers. The Priva Blue system, which is currently used by many customers in the Netherlands, only provides basic alarms for system malfunction etc., whereas the Smart HVAC ΔT tool provides fault detection for all faults occurring in the HVAC system, including the chiller, boiler, air handling unit and its components. Our tool on the other hand is built for a Dutch climate and it helps in the predictive maintenance of all HVAC components, resulting in cheaper operation costs for the customer in the long run.

SWOT analysis

Strengths	Weaknesses
• AI based FDD • Cheaper than manual maintenance contracts • Copyrighted technologies and	• Cooperation with other installation companies for compatibility • Requirement of additional sensors

Opportunities	Threats
• No specific Ai based FDD tool in the market • Requirement to digitize buildings from government	• Possibility of foreign companies entering the Dutch market • Customer may not want to implement unless required by

Marketing

Price

Where in conventional methods customers must pay a separate fee for the BMS system and extra maintenance and fault repair charges with both adding up to around €7-10 / m^2/ year, Smart HVAC ΔT is sold as a standalone FDD application at a lower cost € 0.2 / m^2/ month. The FDD + BMD platform of Kropman is sold at a fee of € 0.5 / m^2/ month.

Promotion

The product will be promoted among customers using social media platforms, TVVL (Dutch Building Association) newsletters, Kropman website as well as building automation and smart building conventions.

Organization

The company will be set up as an independent business unit under Kropman Installatietechniek with the main aim of developing and selling the Smart HVAC ΔT tool. The beta version of the product is developed with another PDEng trainee until the second quarter of 2022. In the third quarter, the company will be set up with the PDEng trainees as employees. Each member works on different modules of the FDD tool. The final version of the product with the software development will require another engineer who will be hired.

Legal structure

The company would be set up as a private limited liability company, *besloten vennootschap (bv)*, with the main shareholder being Kropman Installatietechniek. The board of directors include the PDEng trainees with the board of Kropman Installatietechniek as the supervisory board of directors.

Personnel

The company will initially require two smart building engineers (the two PDEng trainees) and a software engineer. The smart building engineers also act as the directors of the company. One of the engineers will lead the technical development of the product whereas the other will lead the sales and

customer advisory (consulting). In the 2nd year, two additional engineers will be hired to support the team. For research and development, interns will be hired to accomplish specific tasks or goals. Additional engineers for sales and product development will be hired based upon on the revenue and growth of the company from the 3rd year onwards.

Fixed assets

The company will use the workspace (one office room) in one of the buildings of Kropman Installatietechniek. The most relevant assets include workstations and non-tangible property (codebase, algorithms). The workspace would be rented out to the company at a reduced rate compared to the market.

Sales

Sales forecast

The beta version of the product which is to be tested among the customers of Kropman will be released in Q4 2021 and tested among 10 buildings. With the feedback of the customers the product will be updated with the final commercial release aimed at Q3 2022. The aim is to sell the product to 10 customers within the first 12 months of commercial release which will then be extended to 25 customers within the next 12 months.

Customer development

In the first year, the product will be sold to customers of Kropman who use the Kropman BMS software, so that proper user-feedback can be received. Once the product has been successfully tested (beta version as well as final commercial release), the FDD only product will be sold to customers without Kropman BMS system including new building owners (recently constructed buildings), buildings undergoing recommissioning and existing buildings with other BMS software.

Financial analysis

The initial investments for the company are provided by Kropman Installatietechniek. This includes payroll for the employees and equipment. Financial forecasts for three different cases were done: base case, best case and worst case. A profit and loss summary is shown for the base case scenario as shown in Table 12.

The financial model for the base case scenario shows that the company requires external funding of €67,166. For the worst case scenario, external funding of €222,862 is required and for the best case scenario, funding of €20,186 is required.

The value of WACC is assumed to be 12% since the company is a start-up. A very high value of 15% is not used since there are similar companies successfully operating in other parts of the world. In the base case scenario, the pre-money evaluation of the company is €2,137,736. With an additional funding of €67,166, the post-money evaluation becomes € 2,204,902. Kropman then owns 3.05% of the total equity.

In the worst case scenario, the pre-money evaluation of the company is €676,138. With an additional funding of €222,862, the post-money evaluation becomes € 899,000. Kropman then owns 24.7% of the total equity.

Table 12: Profit and Loss summary for base case scenario

Profit and Loss summary		31-Dec-22	31-Dec-23	31-Dec-24	31-Dec-25	31-Dec-26
Sales		300,000	570,000	840,000	1,110,000	1,380,000
Cost of goods sold	(-/-)	(18,000)	(45,000)	(72,000)	(99,000)	(126,000)
Gross Margin		**282,000**	**525,000**	**768,000**	**1,011,000**	**1,254,000**
Personnel cost	(-/-)	(280,000)	(415,000)	(510,000)	(645,000)	(785,000)
Sales & Marketing	(-/-)	(8,000)	(8,000)	(8,000)	(8,000)	(8,000)
General Administration	(-/-)	(12,600)	(18,000)	(23,400)	(28,800)	(36,000)
Research & development	(-/-)	-	-	-	-	-
Other costs	(-/-)	(7,500)	(8,000)	(11,000)	(14,000)	(14,000)
Total cost		**(308,100)**	**(449,000)**	**(552,400)**	**(695,800)**	**(843,000)**
EBITDA		**(26,100)**	**76,000**	**215,600**	**315,200**	**411,000**
Depreciation and amortisation	(-/-)	(5,000)	(5,000)	(5,000)	(5,000)	(5,000)
EBIT		**(31,100)**	**71,000**	**210,600**	**310,200**	**406,000**
Interest	6%	-	(4,030)	(1,309)	-	-
Financial cost		**-**	**(4,030)**	**(1,309)**	**-**	**-**
Earnings before tax (EBT)		**(31,100)**	**66,970**	**209,291**	**310,200**	**406,000**
Tax	25%	7,775	(16,743)	(52,323)	(77,550)	(101,500)
Net result		**(23,325)**	**50,228**	**156,969**	**232,650**	**304,500**
Discounted cash flow		59969.73	38567.26	106764.6	144888.1	170311.2