DATA-DRIVEN AND KNOWLEDGE-ASSISTED MODEL-BASED FRAMEWORKS FOR SUPPORTING FACILITY MAINTENANCE

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

MURAT ALTUN

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF DOCTOR OF PHILOSOPHY
IN
CIVIL ENGINEERING

MARCH 2024

Approval of the thesis:

DATA-DRIVEN AND KNOWLEDGE-ASSISTED MODEL-BASED FRAMEWORKS FOR SUPPORTING FACILITY MAINTENANCE

submitted by **MURAT ALTUN** in partial fulfillment of the requirements for the degree of **Doctor of Philosophy in Civil Engineering, Middle East Technical University** by,

Prof. Dr. Naci Emre Altun
Dean, Graduate School of **Natural and Applied Sciences**

Prof. Dr. Erdem Canbay
Head of the Department, **Civil Engineering**

Assist. Prof. Dr. Aslı Akçamete Güngör
Supervisor, **Civil Engineering, METU**

Examining Committee Members:

Prof. Dr. Rifat Sönmez
Civil Engineering, METU

Assist. Prof. Dr. Aslı Akçamete Güngör
Civil Engineering, METU

Assoc. Prof. Dr. Güzide Atasoy Özcan
Civil Engineering, METU

Assist. Prof. Dr. Saman Aminbakhsh
Civil Engineering, Atılım University

Assist. Prof. Dr. Gözde Bilgin
Civil Engineering, Başkent University

Date: 04.03.2024

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name Last name : Murat Altun

Signature :

ABSTRACT

DATA-DRIVEN AND KNOWLEDGE-ASSISTED MODEL-BASED FRAMEWORKS FOR SUPPORTING FACILITY MAINTENANCE

Altun, Murat
Doctor of Philosophy, Civil Engineering
Supervisor : Asst. Prof. Dr. Aslı Akçamete Güngör

March 2024, 244 pages

Efficient facility maintenance management enhances operational functionality while reducing costs. In practice, however, the lack of (i) historical work order records or their completeness, (ii) updates or complete documentation of facility tasks, and (iii) a sustainable infrastructure makes it difficult to systematically access maintenance information when needed. Moreover, the absence of an intelligent reasoning mechanism extends problem identification and reasoning time. Therefore, this study aims to develop data-driven and knowledge-supported model-based solutions for root-cause reasoning to enhance efficiency in facility maintenance management. In this study, first, an intelligent reasoning approach is proposed for data-driven monitoring to streamline fault reasoning, which combines the maintenance team's expertise with machine learning algorithms in a hybrid intelligence approach to improve the fault reasoning predictions continuously. Hierarchical Neural Networks are developed to group numerous system faults into manageable classification problems, and their prediction capabilities are enhanced through a feedback mechanism developed. Secondly, a BIM-based work order management framework is introduced through visual programming. It links the assets and space to the counterparts in the model and tags observable symptoms, the fault source asset, spatial information, and the impacted assets using symbols and color coding. Using these links in the work order records and standardizing their descriptions, a fault network is created to construct relations between symptoms, fault types, and their assets. When a new work is requested, an analysis approach is proposed to isolate and reason the fault by filtering the network connections utilizing the similarities based on model-derived spatial, systemic, and feature-based relations. The proposed solutions are examined through test cases, and their effectiveness is verified to present the potential of the proposed methods.

Keywords: Building Information Modeling, Facility Maintenance, Fault Reasoning, Hybrid Intelligence, Feedback-enhanced Hierarchical Neural Networks, Fault Network Analysis

ÖZ

TESİS BAKIMINI DESTEKLEMEYE YÖNELİK VERİ ODAKLI VE BİLGİ DESTEKLİ MODEL TABANLI ÇERÇEVELER

Altun, Murat
Doktora, İnşaat Mühendisliği
Tez Yöneticisi: Dr. Öğr. Üyesi Aslı Akçamete Güngör

Mart 2024, 244 sayfa

Etkin tesis bakım yönetimi, çalışma işlevselliğini arttırırken maliyeti azaltmaktadır. Ancak, uygulamada, (i) geçmiş iş emri kayıtlarının tutulmaması yada eksik tutulması, (ii) tesis yineleme bilgilerinin güncel olmaması yada eksik olarak kayıt altına alınması, ve (iii) sürdürülebilir bir altyapının olmaması, bakım bilgilerine ihtiyaç duyulduğunda sistemli erişimi zorlaştırmaktadır. Ayrıca, akıllı bir sebeplendirme mekanizmasının eksikliği, sorunun tespit ve sebeplendirme süresini uzatmaktadır. Bu nedenle, bu çalışma tesis bakım yönetiminin etkinliğini arttırmak için, kök nedenli sebeplendirme için veri odaklı ve bilgi destekli model tabanlı çözümler geliştirmeyi amaçlamaktadır. Çalışmada, ilk olarak, bakım ekibinin uzmanlığını makine öğrenme algoritmalarıyla birleştiren hibrit zeka yaklaşımı ile sorun sebeplendirme tahminlerini sürekli iyileştirmeyi amaçlayan, veri odaklı izleme için akıllı bir sebeplendirme yaklaşımı önerilmektedir. Hiyerarşik Sinir Ağları, birçok sistem hatasını yönetilebilir sınıflandırma problemlerine gruplamak için geliştirilmiştir ve bunların tahmin yetenekleri, geliştirilmiş bir geri bildirim mekanizmasıyla iyileştirilmiştir. İkinci olarak, varlıkları ve mekanları modele bağlayan ve gözlemlenebilir semptomları, sorun kaynağı varlığı, mekansal bilgileri ve etkilenen varlıkları semboller ve renk kodlama kullanarak etiketleyen, YBM ortamında görsel programlama ile iş emri yönetim çerçevesi sunulmuştur. İş emri kayıtlarındaki bu bağlantıları kullanarak ve tanımlamalarını standartlaştırarak, semptomlar, sorun tipleri ve ilgili varlıklar arasındaki ilişkiyi kurmak için bir sorun ağı oluşturulmuştur. Yeni bir onarım işi istendiğinde, model tabanlı mekansal, sistemik ve özellik temelli ilişkileri kullanarak, benzerliklerden ağ bağlantılarını filtreleyen, böylece sorunu izole eden ve sebebini anlayan bir analiz yaklaşımı önerilmiştir. Önerilen çözümler test vakaları ile incelenmiş ve bunların etkinlikleri doğrulanarak potansiyelleri sunulmuştur.

Anahtar Kelimeler: Yapı Bilgi Modellemesi, Tesis Onarımı, Sorun Sebeplendirme, Hibrit Zeka, Geribildirimle İyileştirilmiş Hiyerarşik Sinir Ağları, Sorun Ağı Analizi

To my beloved family,

ACKNOWLEDGMENTS

I want to express my deep gratitude to my supervisor, Asst. Prof. Dr. Aslı Akçamete Güngör for her brilliant guidance and comments, criticism, and encouragement throughout this study. I would also like to thank the examining committee members, Prof. Dr. Rifat Sönmez, Assoc. Dr. Güzide Atasoy Özcan, Asst. Prof. Dr. Saman Aminbakhsh and Asst. Prof. Dr. Gözde Bilgin for their invaluable feedback and contributions to my research. I am also thankful to Assoc. Prof. Dr. Onur Pekcan and Prof. Dr. İrem Dikmen Toker for their valuable ideas.

I want to express my heartfelt thanks to my wife, Kübra Altun, and our beloved child, Ömer Asaf Altun, for their constant support and understanding during this journey. Their love and encouragement have been my rock, helping me through both tough times and successes. I am truly grateful for their sacrifices and unwavering belief in me, as without them, reaching this milestone would not have been possible.

I am grateful to my mother, Şahsiye Altun, my father, Cemal Altun, and my brother, Dr. Mustafa Altun, for their support and positive energy.

I wish to thank my colleagues Dr. Mustafa Göktaş, Dr. Ahmet Bahattin Ersöz, Dr. Yılmaz Emre Sarıçiçek, and Dr. Arda Öcal for all the intellectual help that they have provided.

I hope anyone who reads this study finds valuable information and enjoys it.

TABLE OF CONTENTS

ABSTRACT v

ÖZ…………………. vii

ACKNOWLEDGMENTS x

TABLE OF CONTENTS xi

LIST OF TABLES xv

LIST OF FIGURES xvii

LIST OF ABBREVIATIONS xviii

CHAPTERS

1 INTRODUCTION 1

1.1 Problem statement 3

1.2 Research design 7

2 DEVELOPMENT OF CONCEPTUAL FRAMEWORK ON A REVIEW BASED BIM-ENABLED FACILITY MAINTENANCE MANAGEMENT 15

2.1 Introduction 15

2.2 Challenges in information management in facility maintenance workflow… 18

2.3 Approach 21

2.3.1 Motivation 21

2.3.2 Research scope 22

2.3.3 Review methodology 22

2.4 Model-based maintenance maturity and strategies 25

2.5 A conceptual framework on BIM-enabled facility maintenance management 28
2.5.1 Information requirements and model-based handover 32
2.5.2 Information technologies in model-based facility maintenance 41
2.5.3 Model-based fault management 48
2.6 Discussion and conclusion 54
3 A HYBRID INTELLIGENCE APPROACH FOR DATA-DRIVEN MODEL INTEGRATED FAULT REASONING 59
3.1 Introduction 59
3.2 Background research 60
3.2.1 Hybrid intelligence studies in facility maintenance 60
3.2.2 Fault reasoning in facility maintenance 61
3.3 Research methodology 65
3.3.1 Motivation 65
3.3.2 Fault classification hierarchy 66
3.3.3 A hybrid intelligence approach for fault reasoning 68
3.3.4 Feedback-enhanced Hierarchical Neural Networks 82
3.4 Validation case studies 92
3.4.1 Case Study 1: Simulated multi-zone variable air volume air handling unit 94
3.4.2 Experimental case study 2: experimental single-zone variable air volume air handling unit 114
3.5 Research findings and discussion 130
3.6 Conclusion 140

4 A FRAMEWORK FOR MODEL-BASED WORK ORDER MANAGEMENT AND FAULT NETWORK ANALYSIS APPROACH FOR FAULT REASONING....143

4.1 Introduction....143

4.2 Motivating case studies on work order management in practice....145

4.3 Background research....152

4.4 A model-based work order management framework....154

4.5 A model-based fault network analysis approach for fault reasoning....165

4.6 Validation case studies....169

4.6.1 Mini case study 1: prominence of systematic work request - a pressure problem in a combined boiler....169

4.6.2 Mini case study 2: model integrated work order management in data-driven facility operations – a stuck coil valve problem in AHU operations....172

4.6.3 Mini case study 3: investigation of spatial relations in fault isolation - water leakage problem....177

4.6.4 Mini case study 4: data analytics on the resolution of repeating lighting faults....179

4.7 Discussion and conclusion....181

5 CONCLUSIONS AND FUTURE RESEARCH....185

5.1 Summary of contributions....188

5.2 Limitations of research....190

5.3 Practical implications and recommendations for future studies....193

6 REFERENCES....199

7 APPENDICES....229

A. Statistical information of reviewed articles in Chapter 2....229

B. Asset information parameters of CMMS tools in Chapter 2 230

C. Cross-validation results of validation case studies in Chapter 3 232

D. Work order contents in CMMS tools reviewed in Chapter 4 233

E. Sample examples of model-base work order generation in the Dynamo interface described in Chapter 4 236

CURRICULUM VITAE 241

LIST OF TABLES

TABLES

Table 3.1 Pseudocode of the Proposed Neural Network 86

Table 3.2. Step-by step explanation of the proposed feedback mechanism 92

Table 3.3 Design inputs of case 1 99

Table 3.4 Confusion matrix of the training dataset trained by the initial feedback-enhanced hierarchical model for case 1 105

Table 3.5 Confusion matrix of the test dataset trained by the initial feedback-enhanced hierarchical model for case 1 106

Table 3.6 Confusion matrix of the training dataset for selected fault classes trained by the updated feedback-enhanced hierarchical model for case 1 107

Table 3.7 Confusion matrix of the training dataset trained by the updated feedback-enhanced hierarchical model for case 1 109

Table 3.8 Confusion matrix of the test dataset trained by the updated feedback-enhanced hierarchical model for case 1 110

Table 3.9 Prediction accuracy of different classification algorithms using original dataset for case 1 112

Table 3.10 Prediction accuracy of different classification algorithms using updated training set for case 1 112

Table 3.11 Design inputs for case 2 117

Table 3.12 Confusion matrix of the training dataset trained by the initial feedback-enhanced hierarchical model for case 2 123

Table 3.13. Confusion matrix of the test dataset trained by the initial feedback-enhanced hierarchical model for case 2 124

Table 3.14. Confusion matrix of the training dataset trained by the updated feedback-enhanced hierarchical model for case 2 126

Table 3.15 Confusion matrix of the test dataset trained by the updated feedback-enhanced hierarchical model for case 2 127

Table 3.16. Prediction accuracy of different classification algorithms using original dataset for case 2 129
Table 3.17 Prediction accuracy of different classification algorithms using updated training set for case 2 129
Table 4.1 Statistical information of analyzed work order and request forms 146
Table 4.2 Color codes for description tagging 162
Table 4.3 Symbols for description tagging 163
Table 4.4 Factors determining the prioritization of the fault type inspection 168
Table 4.5 Requested work order of leaking heating coil valve 172
Table 4.6 Filled site feedback module for leaking heating coil valve 173
Table 4.7 Requested work order report for suspicious fault pattern 174
Table 4.8 Requested work order report for the faults in air conditioner. 175
Table 4.9 Filled site feedback module report for stuck outdoor damper 176
Table 4.10 Requested work order report for water leakage issue in the restroom. 177
Tablo A.1 Statistics of categorization for the reviewed article 229
Tablo B.2 Asset information parameters of CMMS tools 230
Table C.3. Cross-validation results of validation case studies 232
Table D.4. Work order request parameters of CMMS Tools 233
Table D.5. Work order parameters of CMMS Tools 234

LIST OF FIGURES

FIGURES

Figure 1.1. Research vision for model-based facility maintenance management and fault reasoning 9
Figure 2.1. Review method of literature studies and their integration to the framework 24
Figure 2.2. A flowchart of model-based facility maintenance management 30
Figure 3.1. Fault hierarchy 67
Figure 3.2. Flowchart of proposed hybrid intelligence approach for fault reasoning 69
Figure 3.3. A typical example of a hierarchical classification model 84
Figure 3.4. (a) 3D view and (b) floor plans of the studied energy resource station building in Ankeny, Iowa 95
Figure 3.5. Fault hierarchy of case 1 98
Figure 3.6. A sample representation of training and test sets in iterative steps 100
Figure 3.7 (a) 3D view of FLEXLAB Test cell X3A and (b) Schematic diagram of single-zone AHU in FLEXLAB 115
Figure 3.8. Fault hierarchy of case 2 118
Figure 4.1. Flowchart of work order request 158
Figure 4.2. Symbolic representation of fault network relations 166
Figure 4.3. Visual representation of fault checklist for the combined boiler 171
Figure E.1. Work order request module: selecting spaces. 236
Figure E.2. Work order request module: selecting symptoms from the checklist 236
Figure E.3. Work order request module: selecting assets from checklist 237
Figure E.4. Work order request module: visual selection of assets 237
Figure E.5. Work order request module: request finalization 238
Figure E.6. Work order site feedback module 239
Figure E.7. Filtered fault network of example case 240

LIST OF ABBREVIATIONS

ABBREVIATIONS

AHU	Air Handling Unit
AI	Artificial Intelligence
AIR	Asset Information Requirements
AR	Augmented Reality
BIM	Building Information Modeling
BPNN	Back-Propagated Neural Networks
CAFM	Computer-Aided Facility Management
CM	Corrective Maintenance
CMMS	Computerized Maintenance Management System
EIR	Exchange Information Requirements
ERP	Enterprise Resource Planning
EA	Exhaust Air
FDD	Fault Detection and Diagnosis
FM	Facility Management
FMMS	Facility Maintenance Management System
FEHNN	Feedback-Enhanced Hierarchical Neural Networks
HNN	Hierarchical Neural Networks
GIS	Geographical Information System
HVAC	Heating Ventilation and Air Conditioning

IFC	Industrial Foundation Classes
IoT	Internet of Things
LCC	Life Cycle Cost
MR	Mixed Reality
NN	Neural Network
OA	Outdoor Air Damper
OIR	Organizational Information Requirements
QR	Quick Response Code
RA	Return Air Damper
RFID	Radio Frequency Identification
PM	Preventive Maintenance
PIR	Project Information Requirements
SVM	Support Vector Machine
VAV	Variable Air Volume
VFD	Variable Frequency Drive
VR	Virtual Reality

CHAPTER 1

INTRODUCTION

The term facility management (FM) encompasses the operational life cycle of a facility, starting from initial operations to the demolishing stage. Throughout this stage, the main aim of the facility management department is to keep the built environment functioning efficiently. Therefore, it consists of various disciplines to manage the facility by considering the roles of people, place, process, and technology (International Facility Management Association, 2018). However, stakeholders often focus on the upfront costs of projects, such as design and construction, during the pre-built stage of a facility. From this point of view, the issues considered in the operational stage of the facility are commonly taken into consideration starting from the end of the construction stage of a facility and sustaining it until the end of its life cycle. Contrarily, the study on life cycle cost (LCC) analysis reveals that, on average, design and construction costs constitute only 15% of a facility's life cycle costs, while FM costs account for approximately 85% (Edirisinghe et al., 2017). Therefore, for design and construction-related faults and any other issues that need to be considered earlier before the construction, the solutions developed in the next steps of the facility life cycle are not as efficient as the ones in the early pre-built stage in terms of operational usability and cost-effectiveness (Altun, 2015). Hence, lifecycle planning in the early stage of facility management gains greater importance to significantly improve facility life cycle costs (LCC) and operational efficiency.

The maintenance and repair tasks comprise at least 65% of the operational cost of the facilities and a significant amount of the time spent (W. Chen et al., 2018). In a facility, naturally, conditional deterioration of the assets and faults in the operational phase of the facility is inevitable. Consequently, either it is detected during a predetermined inspection or notified by occupants, maintenance teams are looking for the problem to find its root and fix its source. However, there are times when information about the problem's location, source, and root-cause is not clear enough to evaluate the situation quickly to solve the problem. In those cases, besides the expertise of the maintenance team on the maintenance task, knowledge of (i) which information is required, (ii) how to gather this information from the facility sources, and (iii) which steps should be followed systematically for the task to extract the information, evaluate it and fix the problem in a reasonable time, plays a crucial role in managing the information efficiently for facility maintenance. Therefore, using this knowledge, identifying possible root-causes of the problem accelerates the evaluation process to find the source of the problem or extracting any beneficial information relevant to the problem facilitates its root finding process.

In contemporary practices, as-built documentation, which encompasses drawings and asset documentation and is typically transmitted to the facility management department, is one of the critical sources used to evaluate the problem. In the operational stage, providing as-built documentation in physical paper formats poses several challenges for facility maintenance management. These challenges include (i) the significant time needed to control the documents for retrieving the relevant information, (ii) the quite complicated process for managing as-built documents to capture facility information, and (iii) the difficulties in following updates in the as-is documents after maintenance and repair tasks during the life cycle. Hence, this approach is susceptible to human errors and necessitates a labor-intensive process. Therefore, computer-aided systems have been developed to digitalize information to either avoid missing information or ease the path to reach the required information for specific purposes. However, considering that FM offers interdisciplinary analysis and management, integrating the different systems for data exchange may create an interoperability problem. On the other hand, Building Information Models (BIM), as a digital representation capturing and exchanging information with those FM systems and information technologies tools, can be used as an information repository to store and deliver as-built information throughout the facility's lifecycle.

As indicated above, for efficient facility maintenance information management, the maintenance team needs to (i) retrieve the required information provided in as-built documents and (ii) keep the history of the previous maintenance tasks for benefiting from contained knowledge in those for the following maintenance steps systematically to find the root and source of the problem and fix it. Therefore, while developing root-cause algorithms to find the source of the problem and to associate the effects of the problem on its environment, BIM can be used as an information repository to retrieve required information from model elements to construct the patterns from those relations. Thus, this facilitates problem solution in facility maintenance.

In this research, we focus on how to utilize BIM for efficient information management in facility maintenance and then develop decision-support solutions interacting with BIM to retrieve information and use its relational intelligent repository, when needed, to facilitate fault reasoning.

1.1 Problem statement

Establishing maintenance strategies is the first step in facility maintenance management. Facility maintenance can be carried out using a variety of maintenance strategies depending on the needs and criticality of assets. In accordance with EN 13306:2010 Maintenance - Maintenance terminology (Standardisation, 2010), maintenance can be basically divided into two main categories: preventive maintenance (PM) and corrective maintenance (CM). PM is generally performed on an asset at a predetermined frequency or when meter readings indicate the asset needs it or the condition of the asset. Meanwhile, maintenance is delayed in CM until a breakdown or fault is found in the asset. In comparison to PM, CM requires more resources. The PM also enhances the asset's lifespan. The statistics show that CM tasks need three times higher cost than preventive ones (Akcamete, 2011). Therefore, facility managers should develop more proactive strategies in facility operations to reduce reactive maintenance and increase scheduled preventive maintenance (Edirisinghe et al., 2013). However, they should have a plan for responding to sudden breakdowns or unexpected maintenance operations. Hence, efficiency in maintenance tasks can be provided by achieving a ratio of 80% (or higher) PM to 20% (or less) CM (Wireman, 2009).

The maintenance strategy of a facility based on preventive maintenance usually offers cost-effective solutions. However, as highlighted above, the maintenance team's reaction to unexpected faults determines the degree of its efficiency. Therefore, it is essential to describe its current maintenance practice to better understand the problem of paper-based and digitalized facility maintenance management approaches (Wireman, 2009). Therefore, in an unscheduled maintenance task, the process begins with the occupant's complaint report for an abnormal condition as a work order request, including space information. The facility manager subsequently proceeds to review the request and allocate the appropriate maintenance team to address the reported complaints and resolve the issue. If the problem is simple and its source can be identified quickly, the maintenance team can fix it within a reasonable time. Conversely, if the problem is complex and poses difficulties in determining its root cause, the maintenance team must consider a variety of possible alternatives in their pursuit of a solution. Therefore, the team visits various locations in the facility one after the other, first looking at what is required and then contacting the maintenance department to obtain the necessary information. Several trips may be required to access instructions, engineering drawings, and tools utilized for the task. If these as-built documents are not available when needed, the team waits until reaching the information. Hence, the lack of support for information leads to an inefficient use of time. The statistics indicate that the idle time of the maintenance team while waiting for relevant maintenance information causes $1.5 billion in waste annually within the United States alone (Gallaher et al., 2004). Each alternative is tested individually and sequentially, or some alternatives are prioritized based on the team's experience until the root cause of the issue is identified. During this process, the capabilities and intuition of the team play a prominent role in solving the problem at a certain time.

However, the more complex the problem is, the more time is needed. It is therefore to be expected that work efficiency decreases, and sometimes the root-causes are not properly recognized and reported, which results in repetitions of the same problem for the next one.

As previously explained, maintenance documents are kept on paper or digital platforms. In a paper-based facility maintenance system, the occupant is expected to call or inform the facility management division to report abnormalities in the space. Then, the manager calls the maintenance team for them to check the problem. During this process, the work order request would probably not be recorded. On the other hand, in the Computerized Maintenance Management System (CMMS), the complaints are entered as work order request tickets, and the manager directly assigns the maintenance team to solve the problem. Until the problem is resolved, the work order seems incomplete, and the facility managers can easily follow the available teams and their locations to direct them to the problem. Moreover, in a paper-based process, effective management occurs when the team has documented solutions or relevant experience. Without such documentation or experience, possible solutions increase exponentially, leading to a time-consuming problem-solving process. In CMMS, maintenance work orders are recorded in the tool, and the drawings are kept in the digital environment to assist in maintenance planning and execution. By searching the historical records, the maintenance team may reach valuable information such as statistics about the failure of the equipment and their possible sources; however, in current practice, a vast number of CMMS tools exist, and they do not have standardized capabilities. Therefore, assuming the availability of the data as mentioned above, only the statistics of the semantic texts recorded by the user as work orders are available. If this data is sufficient to solve the problem, it can be used directly; if not, more intelligent relations between the facility elements are needed. Moreover, the capabilities of CMMS tools to capture the information for supporting maintenance and repair automatically and analyze this data are still limited (W. Chen et al., 2018).

As explained above, facility maintenance practice for root-cause analysis of the problems is quite time-consuming and intuitive. Moreover, it depends on the experience of the maintenance team and their analytical skills to relate the source of the problem with affected associated assets, as well as the capabilities of the computerized tool being used. To reduce the drawbacks and enhance the practice in facility maintenance, BIM-based solutions can be highlighted.

BIM has several potential benefits to facilitate facility maintenance information management and root-cause analysis of the problem for maintenance and repair tasks. Firstly, BIM offers an information repository to link information on the model elements (Eastman et al., 2011). It can organize maintenance documents more systematically to streamline maintenance information flow. Secondly, updating the model according to changes in as-built conditions due to maintenance tasks provides an up-to-date model without missing any information, and the required information can be extracted when needed. Thirdly, it can interact with other computerized databases and information technologies for data exchange (Lavy & Jawadekar, 2014). Fourth, it can provide life cycle information such that information required in the facility operations and maintenance can be analyzed in the design stage of the facility; therefore, any missing and misinterpreted data can be identified and corrected as early as possible. Finally, BIM offers visualization of the model elements to improve the maintenance team's understanding of and communication with as-built conditions (Motamedi et al., 2014).

Besides these potential benefits, BIM has some limitations to support the reasoning mechanism of the facility maintenance tasks. Firstly, BIM offers a flexible and structured information database that contains maintenance-related informative data; however, existing BIM tools have no systematically constructed information database to support facility maintenance tasks (Ensafi & Thabet, 2021). Secondly, BIM only provides linking options for information and model elements. Consequently, it can be used only to extract information from related assets to construct an information network for asset groups, especially for monitoring data for the Internet of Things, or patterns can be recognized manually by using the available information to construct root causes in the tools (Yang & Ergan, 2016a). However, current BIM solutions lack a methodology for automatically building reasoning mechanisms, mainly using asset relations. Thirdly, existing BIM tools do not keep the change history for the updated model (F. Liu et al., 2014). Therefore, the maintenance knowledge/information provided by historical updates is missing when spaces or model elements are removed or replaced. Hence, considering the potential of BIM and its current limitations in practice, a systematic model-based framework using both the current information database and knowledge of the previous maintenance work and changes in the facility is needed to support reasoning mechanisms in the facility maintenance management.

In summary, the absence of maintenance information, inaccurate on-hand information, challenges faced in obtaining the relevant maintenance information within a reasonable time, and lack of a mechanism to interpret the available data to facilitate the root-cause detection of reported faults result in delays in time, increased costs, and inefficient resource utilization. Therefore, these drawbacks motivate this study to develop model-integrated solutions for enabling accurate root-cause detection of the reported fault as decision support tools in facility maintenance.

1.2 Research design

This section outlines the road map for our research, which includes a description of our research objectives (RO), questions (RQ), and methods. As previously explained in the problem statement (in Section 1.1), because of the problems encountered with the maintenance information flow and root-cause identification, time delays occur, costs increase, and resources are not utilized efficiently. To address these issues, we are motivated to examine the potential use of Building Information Modeling in facility maintenance and focus mainly on model-integrated decision support solutions to facilitate the root-cause analysis of problems encountered. Accordingly, the roles of BIM in facility maintenance and its information flow are clearly defined by developing a conceptual framework based on facility maintenance management; then, two different model-integrating decision support solutions for streamlining root-cause of a problem encountered in the facility operations are proposed as illustrated in Figure 1.1.

As summarized above, this research is structured into three distinct sections (chapters from 2 to 4), each dedicated to addressing specific research objectives and questions. A model-based facility maintenance management framework is introduced conceptually (in Chapter 2). Rather than adopting a conventional literature review approach to BIM-integrated facility maintenance studies, we synthesize the previous studies, the capabilities of the commonly utilized computerized maintenance management tools, and our ideas to shape the framework. The main objective of the proposed framework is to design a BIM-enabled facility maintenance management framework mainly based on a literature review for streamlining information management/flow and enhancing fault management, and the secondary one is to identify information needs to sustain the facility maintenance operations timely and cost-efficient throughout its lifecycle. To investigate these objectives, the framework is proposed to address RQ1.

RQ 1: How can a conceptual framework be formed to address the whole process in BIM-driven facility maintenance and fault management?

Taking a holistic view of which maintenance information is required throughout the life cycle of a BIM enabled-facility, how information technologies are integrated with the BIM environment to facilitate data collection, information display, retrieval, and management for supporting maintenance tasks, and focusing on how interoperability issues and integration issues are addressed in model-based management, the focus of the framework is narrowly directed from the maintenance workflow towards model-based fault and work order management and fault detection, diagnosis and reasoning.

While model-based solutions facilitate the maintenance workflow and enhance the process, as stated previously in Section 1.1., specifically, BIM can be used in two different ways to support fault reasoning by linking information and model elements.

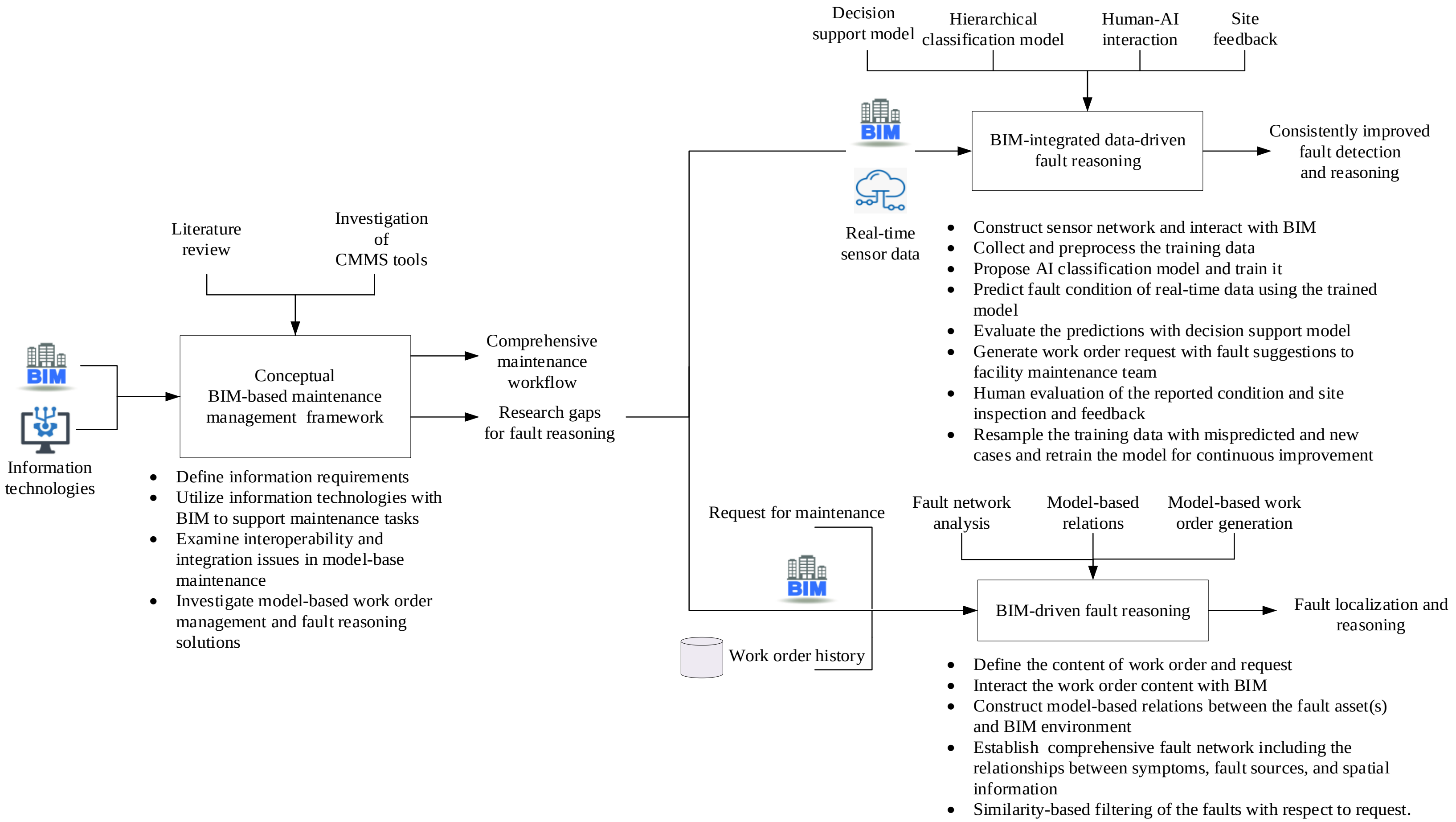


Figure 1.1. Research vision for model-based facility maintenance management and fault reasoning

First, it can be used for information retrieval. To elaborate further, it can extract relevant information of assets from repository to construct on-use network of asset groups for data-driven conditional monitoring. Moreover, BIM can link the maintenance information available in the model to faulty or conditionally deteriorated assets detected by monitoring algorithms for streamlining maintenance tasks (Golabchi et al., 2016). Second, BIM enables an environment to construct root-cause patterns manually using on-hand information (Yang & Ergan, 2016a). Considering the existing capabilities of BIM, in this research, we developed two BIM-integrated decision support tools to facilitate accurate root-cause detection and maintenance tasks. First, maintaining the focus on information retrieval, a hybrid intelligence approach is proposed for the first time in the literature (in Chapter 3). This approach involves the collaboration between artificial intelligence algorithms for fault detection during monitoring and the facility maintenance team. AI predictions assist the maintenance team in making decisions regarding the fault existence and root cause, while the real issues detected by the site team provide feedback to the AI algorithm for retraining it. Hence, both sides support each other to enhance the accuracy of root-cause predictions throughout the lifecycle of fault monitoring and management within a BIM-integrated environment. Second, instead of manually constructing patterns between the problem and its root, we initially proposed a model-based work order management framework where the work order is linked with symptoms, the relevant assets and space in BIM environment to capture and construct the relationship between problem symptoms and its location with the root sources of the fault. Hence, the intelligent information of the assets and spaces are integrated with the symptoms, using spatial, system-based and feature-based relations to create smarter work orders. Hence, contrary to previous studies, the patterns in the work order are recognized automatically using model relations database based on geometric and non-geometric features of assets and spaces. In the next section, we present a fault network analysis approach to identify the root cause of new coming work orders using model-based fault-specific relationships derived from BIM model(s) of the facility and historical work order records. This analysis eliminates some of the possible root alternatives of the problem, prioritizes the relevant ones and sometimes directly addresses the reason for the fault.

As introduced above, the hybrid intelligence approach is developed mainly to improve the accuracy of current root-cause prediction of the faults monitored in living environment and provide continuous improvement in fault management. The feedback from the maintenance team to the AI solution, as well as the accuracy of AI predictions, and their improvement in time, play an important role in upgrading the efficiency of collaboration. Moreover, selection of AI solution and design of fault prediction problem are quite critical for accurate AI predictions. Naturally, in a data-driven system monitoring, first, possible faults in any asset of the system are investigated; then, it is formulated as multiclass classification problem; next, it is trained with representative sampling data; and finally, the trained AI solution predicts the status of the system by testing the real-time data generated. However, in reality, a system consists of multiple assets, each asset has different fault types and even sometimes, the intensity level of this fault may be crucial in decision making to replace the asset with the new one. It is therefore necessary to formulate the problem in quite a lot of classes, but it leads to significant accuracy decrease. In this research, the second objective of this section is to develop a classification approach to divide various faults in a system into manageable units for improving the prediction accuracy. Considering the features of the system, a solution based on hierarchical classification seems more suitable; however, in this classification problem, an incorrect prediction at the higher level adversely influences the accuracy of subsequent predictions at lower levels. In other words, there is a cumulative effect of information loss. These objectives are explored with RQ 2.

RQ 2: How is the performance of AI models on data-driven fault reasoning improved to facilitate fault management throughout the system's life cycle?

While developing an enhanced AI solution, our investigation initially concentrates on how to facilitate interaction between humans and AI models within a data-driven monitoring environment. This interaction aims to enhance fault management and ensure continuous improvement in fault prediction in practical applications. Secondly, our focus shifts to proposing an AI algorithm designed to (i) reframe the fault classification problem into a more manageable format, particularly in systems with diverse fault types across various elements, and (ii) provide accurate predictions for fault causes.

In the light of research objectives and questions, this section introduces a hybrid intelligence framework to provide continuous improvement in fault management and offer a new hierarchical classification method with feedback mechanism called Feedback-enhanced Hierarchical Neural Networks (FEHNN) that predicts fault existence, source, diagnosis, and intensity level respectively to improve the accuracy of the trained model for evaluating the current condition of the elements in a system. The robustness of both hybrid intelligence model and FEHNN are validated with two case studies whereas the hybrid intelligence model is evaluated with updated test data and FEHNN is compared with different AI methods and alternative neural networks.

As explained above, the last section focuses on utilization of model-based relations in the pattern construction for facilitating the root-cause identification of encountered issues. Due to the absence of interpretable model-based historical records of work orders and manually constructed root-cause patterns in the current literature (Yang & Ergan, 2016b), we need first to create an environment for interacting the work order records with the contents of the model to construct the relations; then find a solution to evaluate these relations for streamlining the root of the encountered problem. Therefore, the first objective of the third section is to develop a systematic model-based work order management framework for constructing patterns between symptoms and the source of the fault, space and the impacted assets from the fault using their model-based relations. After constructing model-based framework for work order management and collecting the work order with detected relations between the relevant items, the second objective of this section is to develop a network analysis approach that analyzes possible alternative roots of the problem and provides decision support to find its real root. These objectives are questioned with RQ3.

RQ 3: How can a BIM-enabled work order management framework be developed to facilitate network analysis of faults and enhance model-based fault reasoning?

While developing a model-based solution, the investigation focuses on maximizing the information gained from previous work orders and available information by examining the formation of work order content and requests. Additionally, the integration of this content into a BIM environment to enhance the intelligence of a work order management framework is explored. Furthermore, the utilization of BIM to establish relationships between assets during faulty conditions is examined. Finally, the investigation explores the development of a structured network of relationships between symptoms, fault sources, and spatial information to enhance the accuracy of prediction for reasoning mechanisms.

In this section, first, we propose a model-based framework for work order management in BIM environment coded via visual programming Dynamo. The framework consists of three modules: work request, work order analysis and management and work order site module. While information for maintenance request and site feedback for the work order is collected via the interface of the framework, data analysis and the management of work orders are offered as part of the background support. Using on-hand information provided in work order request and historical records of work orders, with smart tags, network relations are automatically constructed in the encoded environment between symptom, space, fault root and other impacted assets. These relations are built upon physical, spatial, and systemic relations, as well as similarity of asset's features. After that, the steps of the framework are first tested with use case studies for verification; then, how the model constructs the relations using the given smart information is validated using cases of each relation type. In the framework, each work order is stored in the information repository, including its work request and site feedback details and possible child work orders. Hence, using these records and the relationships detected in each order, a comprehensive fault network is constructed. When new work is requested, the possible roots of the problem are evaluated using the network. In this research, a fault network analysis approach is proposed to (i) filter the roots based on the reported symptom and space; then (ii) apply fault measure rule(s) to isolate fault source if available; and (iii) finally prioritize the remaining roots according to their similarity, critically and frequency. Hence, possible fault sources of the problem in a prior list are reported to the site team to facilitate the task. The efficiency of the proposed approach is validated with work order cases.

In summary, this research is organized in five chapters. First, Chapter 1 presents the description of the problem and design of the research including the research objectives, question, methods, and scope of the thesis. Chapter 2 describes a conceptual model-based facility maintenance management framework developed by inspiration from literature studies on this topic, especially focusing on fault management. Chapter 3 introduces a hybrid intelligence framework in BIM-integrated data-driven facility monitoring to provide continuous improvement in fault management and offer a new hierarchical classification method with feedback mechanism called Feedback-Enhanced Neural Networks to handle possible information loss. In Chapter 4, a model-based work order management framework is proposed to construct the relationship between symptoms and space and the source of the fault using the framework's enriched content. Additionally, we offer a fault network analysis approach based on the work orders created by using fault-specific and model-based relationships to isolate the reason for abnormalities in the reported environment. Finally, Chapter 5 brings the main research findings to a close, discusses the limitations of the research, and proposes potential directions for future research

CHAPTER 2

DEVELOPMENT OF CONCEPTUAL FRAMEWORK ON A REVIEW BASED BIM-ENABLED FACILITY MAINTENANCE MANAGEMENT

2.1 Introduction

Maintenance plays a crucial role in ensuring the longevity and functionality of a facility (Teicholz, 2018). Maintenance management encompasses structured methods and frameworks that facilitate the planning, execution, and monitoring of maintenance activities. These systems represent central pillars in facility management, providing essential support for the efficient oversight, tracking, and prioritization of maintenance tasks. Their significance extends beyond routine upkeep, encompassing critical functions such as minimizing downtime, managing costs, optimizing resource allocation, and ensuring the safety and reliability of the facilities they govern (Fraser, 2014).

Encompassing a diverse range of functions, Facility Maintenance Management Systems (FMMS) oversee asset tracking and management, coordinate work orders, schedule preventive maintenance tasks, manage comprehensive documentation and data, generate insightful reports and analytics, efficiently handle inventory, coordinate with vendors, and ensure compliance with regulatory standards (Hu et al., 2018).

Paper-based facility maintenance management systems suffer from several drawbacks in today's digital age. Firstly, they are prone to human error, as manual data entry increases the likelihood of mistakes in recording maintenance activities, scheduling tasks, and tracking inventory (Bortolini et al., 2016). Moreover, paper-based systems lack real-time updates and accessibility, making it challenging to keep all stakeholders informed and up-to-date on maintenance schedules and issues. Additionally, storing and organizing paper records can be cumbersome and inefficient, leading to difficulties in retrieving information when needed (Smillie et al., 1988). Therefore, paper-based systems often result in slower response times, decreased productivity, and higher operational costs due to their inherent limitations. On the other hand, computerized facility maintenance management systems offer numerous advantages over paper-based systems, including improved efficiency, accuracy, and accessibility. However, they also have their drawbacks. They sometimes struggle with integrating data from different sources that cause compatibility problems with hardware or other software applications (Moreno et al., 2022). These issues can disrupt maintenance operations and lead to downtime, negatively impacting facility performance and service delivery. Visualization and collaboration are limited (Jiang et al., 2017), impacting teams' ability to identify issues and work together effectively. Moreover, these systems may be complex, difficult to customize, and lack scalability, potentially increasing maintenance costs and inefficiencies over time (Alshokry et al., 2021). Additionally, computerized systems may require substantial initial investment in software licenses, hardware infrastructure, and staff training. Moreover, there may be challenges in data security and privacy, particularly concerning sensitive maintenance records and asset information.

Introducing a BIM-based facility maintenance management system addresses many of these drawbacks while offering a host of potential additive values. BIM enables improved data integration, allowing maintenance teams to access comprehensive and up-to-date information about facility assets and maintenance history (Gao & Pishdad-Bozorgi, 2018). This facilitates proactive facility maintenance by enabling predictive analysis and preemptive maintenance planning (Shalabi & Turkan, 2017). Moreover, BIM facilitates life cycle management by incorporating maintenance plans into the design stage (Heaton et al., 2019), ensuring that maintenance considerations are integrated from the outset. Asset performance monitoring is enhanced through BIM's ability to provide real-time data on asset condition and performance (Lu et al., 2020). Furthermore, BIM-based systems offer enhanced visualization capabilities, allowing maintenance teams to better understand the facility layout and identify maintenance requirements more efficiently (Akcamete et al., 2010). Collaboration and communication are also improved through BIM, as stakeholders can easily share information and coordinate activities within a centralized platform (Korpela et al., 2015; Valdepeñas et al., 2020). Maintenance workflows are streamlined through automated processes and centralized data management, reducing administrative burden and improving overall efficiency. Additionally, BIM-based systems offer future scalability and flexibility, allowing for easier adaptation to changes in facility requirements or technological advancements.

Information management in facility maintenance is significantly enhanced through BIM-based systems (Wijekoon et al., 2020). BIM serves as a comprehensive information repository, housing all relevant data pertaining to facility assets, maintenance schedules, work orders, and historical records. This centralized approach to information management ensures data integrity, consistency, and accessibility, minimizing the risk of errors and redundancies inherent in manual or disparate systems. Maintenance teams can easily retrieve relevant information, track asset history, and analyze trends to make informed decisions regarding maintenance strategies and resource allocation. BIM also enables the integration of various data sources, including IoT sensors, equipment diagnostics, and performance metrics, providing a holistic view of facility operations for more effective management.

In the prior studies, the researchers explored different aspects of facility maintenance and information management interacting with BIM; however, there remains a critical need for a comprehensive review that synthesizes and integrates these diverse findings to provide a holistic understanding of model-based maintenance management. Therefore, in this research, we explore a conceptual framework for model-based maintenance management, mainly focusing on challenges in information and fault management.

2.2 Challenges in information management in facility maintenance workflow

Effective information management is crucial in facility maintenance workflows as it enhances the organization, accessibility, and utilization of critical data. Properly managed information facilitates streamlined communication, ensuring that maintenance teams have timely access to essential documentation, schedules, and historical records. This not only promotes efficient decision-making but also minimizes downtime by enabling proactive maintenance planning and swift response to issues. Additionally, comprehensive information management supports compliance with regulatory requirements, enhances collaboration among team members, and contributes to the overall optimization of maintenance processes, fostering a more cost-effective and reliable facility management system.

Ensuring effective information management proves to be challenging in practice. According to research conducted by the National Institute of Standards and Technology (Gallaher et al., 2004), the process of verifying or validating information for accurate representation incurs substantial time and costs, totaling $4.8 billion in labor charges annually. Once the information is obtained, around $613 million is allocated for operations and maintenance engineers to convert the verified data into an accessible and usable format for staff members performing their tasks. Moreover, due to a significant portion of information being stored in paper format, accessing it consumes considerable time, leading to information delays. Therefore, these delays, causing workers to wait idly for information to address maintenance issues, incurred a cost of nearly $1.5 billion in 2002. Furthermore, when dealing with insufficient information, operations and maintenance staff often need to revisit maintenance problems for correct resolution, incurring additional costs. The annual cost of inadequate interoperability for these staff members amounts to $6.9 billion, resulting in $1.92 per square meter per year.

Throughout the maintenance workflow, workers encounter various information-related challenges that lead to inefficiencies when resolving issues. In accordance with Hicks' analysis of the factors leading to information waste (Hicks, 2007), these inefficiencies can be recognized through four specific pathways: (i) essential information being unavailable, (ii) dealing with an overload of information, (iii) encountering inaccuracies in information, and (iv) facing difficulties in accessing the relevant information. Initially, if essential information is not found in current documents stemming from gaps in data collection and incomplete or inadequate recording (Ensafi et al., 2023), workers must rely on reactive responses based on their knowledge and experience to obtain information and address the issues. In addition, higher personnel turnover in this practice poses a significant challenge to preserving and transferring critical knowledge (Moon et al., 2022). The loss of experienced individuals diminishes institutional knowledge, impacting the information-centric practices that rely on historical insights. Limited training and orientation exacerbate the situation, hindering the effective utilization of information in resolving maintenance issues. The risk of safety incidents increases as new workers might lack the necessary information to address equipment malfunctions safely. Knowledge transfer challenges are heightened, impeding the seamless sharing of best practices and lessons learned. As a result, it reduces process efficiency, increases asset downtime, and potentially introduces safety risks. Moreover, dealing with excessive amounts of data from diverse maintenance sources poses a challenge for workers in efficiently managing and retrieving relevant information. Particularly in conditions where the searchability of maintenance information source is limited such as scanned or paper-based documentation, the surplus of data can result in information fatigue, impeding maintenance personnel from promptly accessing vital information.

When the process of handing over the relevant maintenance information lacks clear definition and accountability, manual data entries in facility maintenance workflows, often relying on unstructured data, tend to amplify human errors and compromise the quality of information. Therefore, without a proper feedback mechanism and sufficient training for addressing errors in the process and awareness among personnel regarding the importance of accurate data entry and maintenance of records, these entries might introduce inaccuracies and inconsistencies, undermining the reliability of information for maintenance tasks. Furthermore, when there is a lack of regular updates to maintenance information, encompassing manuals, equipment specifications, standard operating procedures, and changes in the facility's actual state, discrepancies can arise. This inconsistency diminishes the reliability of maintenance planning, as the documented information may no longer accurately reflect the current condition of the facilities.

Accessing maintenance information within the system can be challenging, even when it is readily available. In a complex workflow, maintenance information might be dispersed across various stages, complicating the understanding of task flow and dependencies for workers and hindering the prompt location of relevant information. The need to follow intricate procedures or deal with convoluted processes can lead to delays in accessing critical information. Moreover, when maintenance tasks are outsourced to different vendors or information is dispersed across various platforms, the absence of standardized practices in data entry and information management processes leads to inconsistencies and errors. This situation causes interoperability problems, jeopardizing data integrity and hindering seamless information exchange. Consequently, it may necessitate costly and time-consuming custom integrations, making it challenging to achieve a unified view of maintenance data. It results in isolated repositories, limiting the seamless flow of information. Hence, this decentralization hampers accessibility to information repositories and introduces redundancy in maintenance environment where similar information is stored on various platforms. Furthermore, inadequate communication channels and collaboration platforms between maintenance stakeholders can lead to misunderstandings, delayed responses, and increased downtime.

2.3 Approach

2.3.1 Motivation

Our departure from a traditional literature review on BIM-based facility maintenance management systems signals a strategic move toward innovation. Rather than dwelling solely on existing practical approaches, we aim to develop a conceptual framework. This shift is grounded in a deliberate emphasis on the profound significance of information management throughout the maintenance workflow.

The motivation for this shift arises from the recognition that existing facility maintenance management practices lack a unifying and systematic model. While some approaches focus on specific aspects of model-based facility maintenance, they fall short of a cohesive and integrated framework that maximizes the benefits BIM offers throughout the maintenance process.

By highlighting this absence, our objective is to underscore the significance of establishing a conceptual framework. This framework will transcend the limitations of current practices, providing a structured and unified model for leveraging the transformative potential of BIM technology in facility maintenance. The emphasis on information management within the proposed framework acknowledges its crucial role in ensuring the seamless integration of BIM throughout the maintenance workflow.

Through a rigorous examination, analysis, and integration of relevant literature studies, our goal is to contribute a conceptual framework that not only addresses the current gap but also introduces a fresh perspective for holistic and efficient BIM-based facility maintenance management practices.

2.3.2 Research scope

This research endeavors to explore the influential factors shaping the model-based facility maintenance workflow with the goal of proposing a robust conceptual framework for enhanced efficiency. Emphasizing the pivotal role of information management within this framework, our approach involves a comprehensive examination of literature studies, specifically adopting an information-centric workflow. Within this context, our focus is directed towards understanding the essential information required for optimizing the workflow, as well as devising information technologies for collecting and evaluating this information. Recognizing the critical impact of information on the overall process, our aim is to develop insights that contribute to the refinement and optimization of facility maintenance workflow. Moreover, to offer a more precise and nuanced viewpoint, our framework intentionally restricts its scope by taking faulty conditions into account. This approach places distinct emphasis on essential dimensions such as model-based fault management, work order management, and root-cause analysis. This purposeful narrowing of focus enables a comprehensive exploration of these specific areas, leading to the advancement of a theoretical framework designed to address the intricacies of model-based facility maintenance workflow across diverse conditions.

Recognizing the significance of analyzing model-based maintenance maturity and strategies to enhance our comprehension of the framework, we initiated the process by proposing a model-based maturity model integrated with maintenance strategies. This initial step laid the foundation for the ongoing development of the comprehensive framework.

2.3.3 Review methodology

In our research, we thoroughly investigated literature studies focusing on the integration of Building Information Modeling in facilities maintenance. The aim was to consolidate the findings of these studies and to develop a comprehensive framework for the management of facilities maintenance, utilizing models as a foundation. Drawing inspiration from Gao and Pishdad-Bozorgi (2019), the review process, as depicted in Figure 2.1., involves a sequential approach encompassing four key steps. Firstly, it entails delineating the research scope for model-based facilities maintenance management and strategically selecting keywords for exploring relevant articles within specified databases. Secondly, the process involves the systematic collection of articles along with their abstracts. The third step requires the application of content analysis to filter out the irrelevant articles, categorize them based on their relevance to model-based maintenance issues, and articulate their specific contributions and findings within the defined research scope. Lastly, the synthesized contributions play a pivotal role in shaping the development of the framework, and the ensuing discussion revolves around these integrated insights.

To collect the articles , we first conducted a keyword search in Web of Science and Scopus databases, restraining the publication year of the article between 2005 and 2023. 548 articles whose title or abstract containing “BIM” and “maintenance” were extracted from the databases and consolidated them into a separate database, preserving the contextual relevance, along with their respective abstracts. After that, we reviewed 423 abstracts of the collected articles and eliminated those deemed irrelevant. The remaining articles were systematically classified based on their relevance to model-based facility maintenance into four categories: (i) addressing general issues related to BIM-enabled facility maintenance management, encompassing information requirements, benefits, values, challenges, maintenance strategies, accessibility, practical maintenance actions, safety, and legal issues; (ii) focusing on data integration and model-based maintenance systems, covering developments in literature, case studies in diverse areas, integration of information technologies into model-based facility maintenance, and interoperability issues; (iii) delving into model-based fault management, including data analysis and management, work order management, conditional assessments, fault detection and diagnosis, and root-cause analysis; and (iv) capturing secondary sources that contribute to the framework, such as model-based facility management studies,

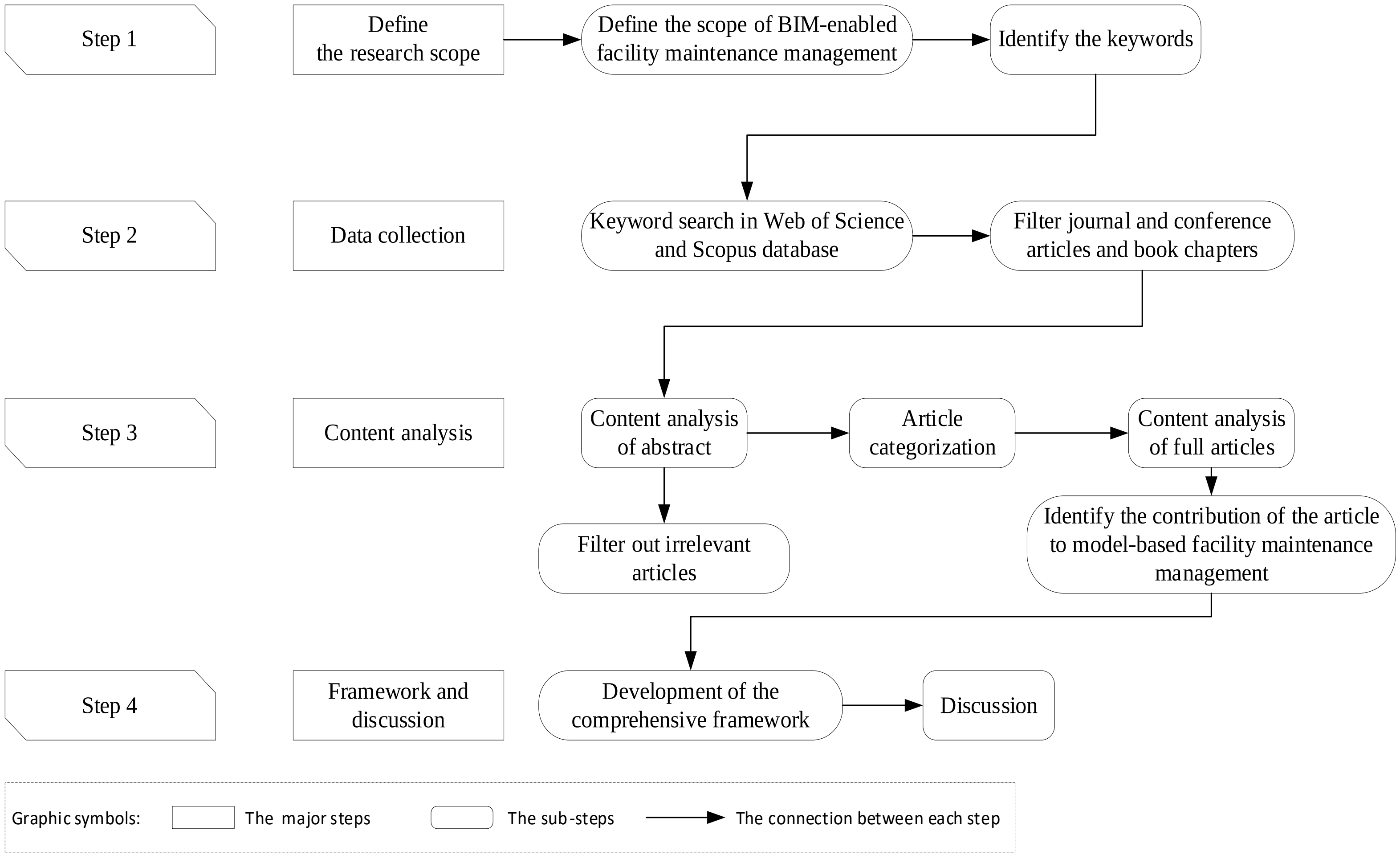


Figure 2.1. Review method of literature studies and their integration to the framework

digital twin applications, location-based solutions, and asset and space management. The statistical details of the studies are outlined in Appendix A. A thorough examination was conducted on the articles belonging to the first three categories to extract their findings and assess their potential contribution to our framework. Meanwhile, the remaining articles were scanned to identify additional insights that could enhance the value of our proposed solution.

The identical procedure was applied with a heightened focus on fault management. Therefore, additional keywords were stipulated to broaden the search scope: “BIM” and “work order”, “BIM” and “fault detection and diagnosis”. Articles identified through these searches were initially cross-referenced with those from the previous search, and only new and relevant ones were incorporated into the evaluation and reported in Appendix A.

2.4 Model-based maintenance maturity and strategies

Inspired by maturity models of process management (Rosemann & De Bruin, 2005), BIM (Alankarage et al., 2023), data governance (G. Cheng et al., 2017), and capability (Gökalp et al., 2022), the maturity of BIM-based facility maintenance management follows a progressive journey, evolving from initial ad-hoc state to optimized level of excellence. Initially, maintenance activities are reactive, unstructured, driven by immediate needs and reliant on the knowledge of the maintenance personnel. Since there is a lack of formalized processes and standards for gathering, managing, and exchanging information within an organization, information needs are often addressed on a case-by-case basis, without a systematic approach or clear guidelines. As the documents and 2D CAD drawings are independently prepared, there is a restricted collaboration and communication scope regarding the collection, storage, and sharing of data. This situation leads to the creation of data silos, diminishing the overall operational visibility. Moreover, communication tends to be informal, relying heavily on personal interactions and individual preferences rather than established protocols. As a result, information may be scattered across various sources, making it challenging to access and use effectively. Data entry processes are likely manual and involve redundant efforts. Maintenance personnel manually record information on paper or in isolated digital documents, leading to the risk of errors, loss of data, and inefficiencies. However, as maintenance costs rise, operational disruptions occur, and organizational limitations become apparent, the owners and operators recognize the prominence of more systematic organization, especially for repetitive maintenance tasks for critical assets whose disruption fully or partially impacts the operations of the facility. Therefore, growing awareness leads the organization to establish systematic solutions such as introducing preventive measures and routine inspection for those assets and collecting and storing information required for their maintenance. This leads to scheduled maintenance inspections for critical assets, mitigating the risk of unexpected breakdowns. Moreover, despite limited collaboration among stakeholders initially, early efforts facilitate basic 3D modeling of the facility, integrating geometric and semantic data. Consequently, raw facility data gradually transforms into more organized information. Information is still somewhat fragmented, but efforts are made to consolidate and centralize data repositories. Hence, basic maintenance management systems such as spreadsheets are introduced to track maintenance activities, especially for critical assets. However, processes remain predominantly manual and reactive at this stage.

With increasing capabilities, this approach permeates the entire organization from key assets; hence, the maintenance processes are systematically formalized, and roles and responsibilities, clear documentation, and standardized practices are defined to ensure consistency in performing the maintenance tasks. Within more collaborative efforts of the stakeholders in 3D modeling, BIM standards are adopted to create the model, and data exchange standards such as Industrial Foundation Classes (IFC) and Construction Operations Building Information Exchange (COBie) are introduced to facilitate the interoperability between the models and integration of other tools (East, 2011). It establishes standardized processes for collecting, organizing, and analyzing facility data. Data management systems are implemented to ensure consistency and reliability in data capture. Knowledge about facility assets, their condition, and maintenance requirements becomes more structured and accessible for preventative maintenance.

Once the model-based maintenance processes have been standardized, proactive solutions are then developed to enhance the maintenance efficiency, utilizing control mechanisms and performance metrics on the formalized procedures in practice. By interacting information technologies to monitor the systems and track the assets and computerized maintenance management system and computerized facility management systems with the standardized model, key performance indicators are defined, and real-time data and historical records obtained from various sources of information are analyzed to gain insight about the performance and maintenance needs of the facility, and condition of the assets. Knowledge obtained based on a deep understanding of asset behavior and performance trends is evaluated to propose proactive maintenance strategies to update model-integrated maintenance schedules, prioritization, and procedures.

Model-based maintenance management is fully matured with the development of feedback mechanisms to facilitate continuous improvement, addressing not only current maintenance needs but also anticipating future demands. At this stage, a visionary maintenance approach rooted in wisdom-based decision-making is adopted. This approach leverages knowledge for strategic decision-making and utilizes data insights to drive continuous improvement in operations, moving away from corrective solutions to maintenance issues on a daily basis. Therefore, AI-driven predictive solutions are embraced to both monitor and predict the condition of assets and guide optimization decision-making. Under this paradigm, continuous improvement takes precedence, and maintenance strategies are dynamically adjusted based on real-time data and feedback. Moreover, while generating the digital model, collaborative efforts are maximized between stakeholders that integrate all aspects of the project lifecycle and allow them to work simultaneously to create a model. Full interoperability is provided while information is shared between the stakeholders, and information technologies are integrated into the model. Hence, a flexible and adaptable information environment for maintenance issues is provided within full lifecycle BIM integration.

In determining the maintenance strategy in a facility, several factors are investigated (Patil et al., 2022; Zaim et al., 2012). First and foremost is the criticality of assets, which guides prioritization efforts, ensuring resources are allocated to maintain vital components essential for operational functionality and continuity. Second, adherence to regulatory standards and practices is essential to ensure compliance. Third, prioritizing the safety and comfort of occupants and considering the impact on the community are critical, necessitating strict adherence to regulations to uphold a positive reputation. Fourth, cost considerations balance expenditures with the desired level of service, ensuring resource allocation is optimized. Fifth, risk tolerance informs decisions regarding the extent of maintenance required to mitigate potential risks while aligning with the facility's organizational objectives. Moreover, the availability of skilled personnel influences the complexity and execution of maintenance tasks, while technological integration facilitates streamlined processes. Furthermore, equipment complexity and worker acceptance necessitate tailored strategies to address specific technical challenges effectively.

2.5 A conceptual framework on BIM-enabled facility maintenance management

In this study, a conceptual framework is developed to offer holistic and efficient BIM-based facility maintenance management inferred and synthesized from the insights gathered from previous studies and practices. This model-driven framework leverages the potential of BIM to tackle the challenges of information management in the maintenance workflow. Initially focusing on maintenance management, it gradually narrows its scope to model-based work order management and fault detection and reasoning that maximizes the information gained to reduce downtime and maintenance costs of the facility. The model-based framework consists of the following main sections: (i) information requirements, (ii) data handover standardization and interoperability of facility maintenance solutions, (iii) integration of information technologies to maintenance workflow, data, and information management, and (iv) model-based fault management including work order management and fault detection, diagnosis and reasoning.

As depicted in the flowchart shown in Figure 2.2, in the initial design stage of the facility, the FM department collaborates with the design department to convey their information needs and expectations to them. Within the integration of BIM into the lifecycle of the facility, these interactions are more concretely facilitated and streamlined. To mitigate additional maintenance and repair needs or operational inefficiencies throughout the facility's remaining lifetime, design details susceptible to faults and inefficiencies, as well as difficulties in navigating maintenance routes and accessing maintained assets and spaces, are analyzed within the model-driven environment. Necessary updates are then made according to the identified needs. The FM department is responsible for organizing the model-based information requirements. They determine (i) the purposes for which maintenance information is needed, (ii) the specific information entries required to meet the needs of those purposes, (iii) the format and methodology for collecting the data, and (iv) designate the responsible party as the data provider. The procedure and data format for preparing and integrating data into the models follow standardized to streamline the handover process. The promotion of widely accepted data exchange formats such as Industrial Foundation Classes (IFC) and Construction Operations Building Information Exchange (COBie) is strongly advocated to enhance interoperability during data exchange. In instances where these standards may be insufficient, data exchange is facilitated through the utilization of custom-made, generalized templates, achieved by leveraging visual programming tools that interact with the facility model or Application Programming Interfaces (APIs). Geometric and some non-geometric data are shared, sourced from design and construction models, and tailored to fit the facility model. At the handover stage, additional facility-specific operational details are directly incorporated into the model. This framework utilizes a hybrid maintenance strategy customized to individual assets or systems, dynamically adjusting to evolving working conditions. This strategy not only addresses maintenance needs but also influences information requirements accordingly, especially for maintenance scheduling and work order creation.

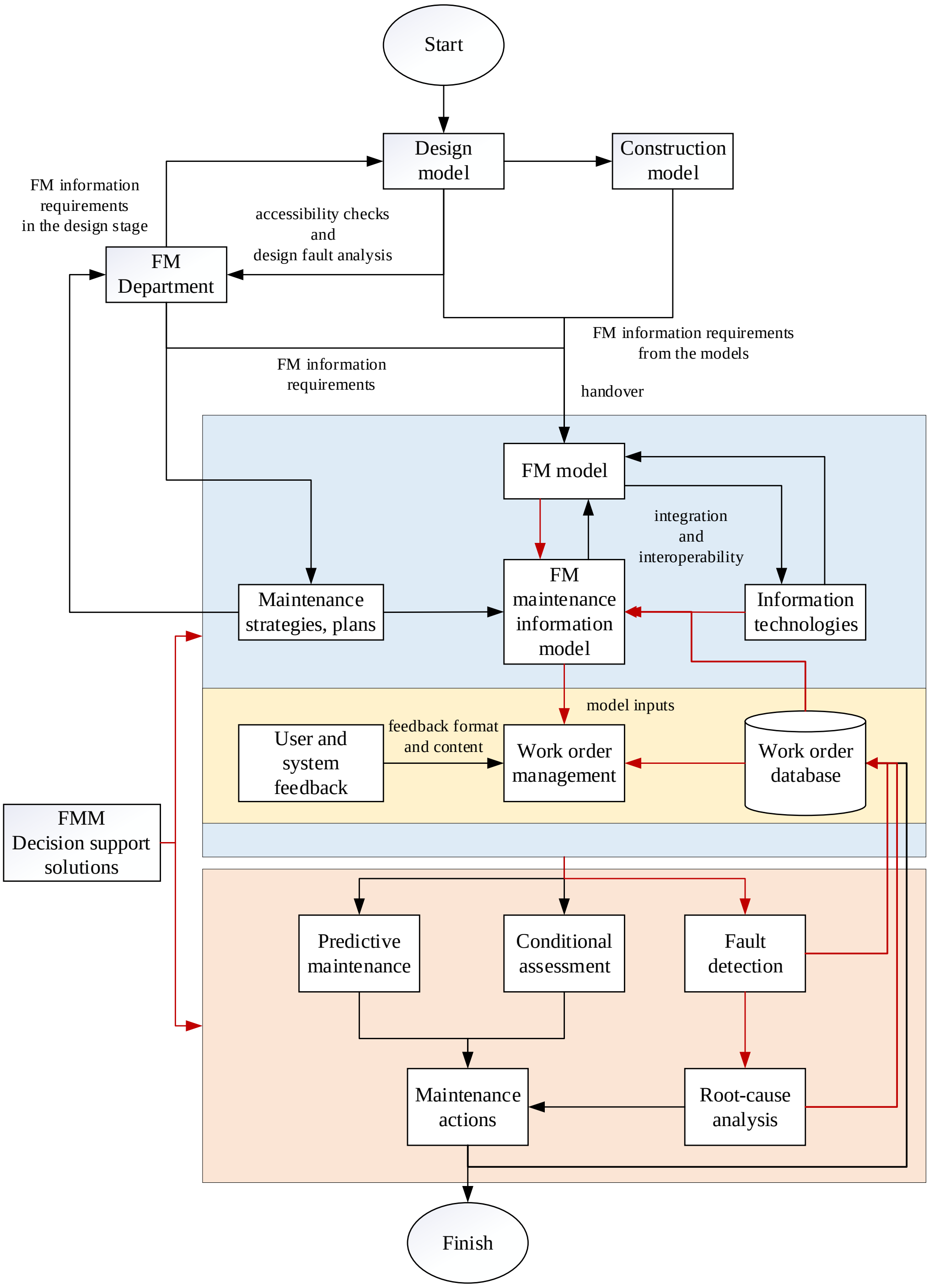


Figure 2.2. A flowchart of model-based facility maintenance management

Once the facility management model is completed, it is synchronized with Computer-Aided Facility Management (CAFM) and Computerized Maintenance Management System in a common data environment to enhance space utilization efficiency and manage the maintenance workflow of the assets, respectively. Information technologies have been seamlessly integrated into this environment to streamline data collection, information display, retrieval, and management. Interacting with BIM, these technologies optimize workflow by bridging static models and systems with dynamic IT solutions. Through this synergy, maintenance plans and strategies are dynamically communicated, culminating in a comprehensive maintenance information environment. The accumulated data and information from the assets are consolidated to be comprehensively understood and utilized for enhancing model-based maintenance within the contribution of decision-support solutions. Therefore, this environment serves multiple purposes to optimize the maintenance workflow:

(i) supporting maintenance tasks and visual analytics with enhanced visualization capabilities.
(ii) tracking assets to access and examine spatial and descriptive documentation, historical records, maintenance schedule information, and monitoring system and equipment performance for predictive and condition-based maintenance decision-making.
(iii) facilitating training and practical efforts, including easy-to-track model-based procedures, virtual and remote collaboration, accessibility of the maintenance area, path planning, and indoor navigation access to information on-site.
(iv) detecting faults, reporting them with work orders, and analyzing root causes to facilitate maintenance tasks.

Facilities are dynamic living environments that require adaptation to changing environmental conditions to maintain functionality and operate efficiently. As the collected data and information are comprehended, decision support solutions are strategically updated to leverage the insights for continual improvement of the facility's functionality. This framework strongly supports continuous improvement of the maintenance workflow within closed-loop feedback. As illustrated in the flowchart, data is collected in a structured format whenever possible. When determining its content, a participatory feedback mechanism is employed, ensuring that the feedback of maintenance personnel who utilize the data and its output in decision-making or application are evaluated. Moreover, feedback provided for conditional monitoring, predictive condition of the assets, and fault management, including fault detection and reasoning and maintenance actions, are collected in a database, and used for updating the predictive models.

In the following sections, the details of the framework are sequentially examined and discussed.

2.5.1 Information requirements and model-based handover

Ensuring that information shared during handover is accurate, relevant, and complete is essential for its efficient use in facility operations and maintenance (Ghosh et al., 2015). However, the current handover process suffers from delays, lack of structure, and inconsistencies in documentation. Before commissioning, facility management personnel often lack awareness of the equipment and systems they're responsible for, leading to manual data entry after handover (Cavka et al., 2017b). To address potential disruptions in information flow, this framework adopts the international standards ISO 19650 (X. Pan et al., 2024). It aims to organize the handover procedure and content, making it easier to collect relevant information and manage it using models (Malla et al., 2024). To systematically define information requirements, it is essential to first clarify the strategic maintenance objectives and goals of the facility. This entails understanding the overarching vision and mission of the facility in relation to its assets and infrastructure, thereby laying the foundation for precise identification and prioritization of information needs. Within this framework, a proactive solution, enriched by feedback mechanisms, is implemented to optimize a dynamic hybrid maintenance strategy. This approach integrates information technologies to facilitate model-based facility maintenance, enhancing efficiency and effectiveness in managing assets and infrastructure.

A methodical sequence of steps characterizes the information requirements for BIM-based facility maintenance. It begins with defining Organizational Information Requirements (OIR), which encompass the data and information needs of the entire facility, particularly concerning maintenance strategies and practices. OIR identifies the overarching objectives and goals of maintenance activities in the facility. In this framework, it defines the strategic direction for maintenance, such as utilizing information technologies and structured information and visualization in the BIM environment to streamline maintenance information workflow, implementing predictive maintenance within information technologies to optimize asset performance and detect the faults and its reason, visualize the maintenance accessibility and path, and fault patterns to enhance comprehension, applying an asset or system specific hybrid maintenance strategy. Additionally, OIR entails establishing performance metrics to measure the effectiveness and efficiency of maintenance operations at the organizational level. By aligning organizational information requirements with the framework, facilities ensure that the necessary data and information are available to support decision-making, drive continuous improvement, and achieve maintenance objectives effectively. Therefore, to complement OIR, Asset Information Requirements (AIR) are defined to delineate the specific data and information needs related to individual assets or systems within the facility including asset identification, specifications, maintenance history, performance criteria, operating manuals, and other pertinent information required for maintenance, operations, and decision-making. It ensures that the detailed information needed for the maintenance of each asset is identified, structured, and accessible within the BIM environment. In contrast to the specific focus of AIR on assets, Project Information Requirements (PIR) outline the data and information required to plan, execute, and monitor maintenance activities effectively within a facility's lifecycle, typically during the design, construction, and handover phases. PIR guide the development and exchange of information throughout the project lifecycle, ensuring that stakeholders have access to the necessary data and information to support decision-making and project execution within the BIM environment. After defining the information requirements, the next step is to plan how to collect the data. Exchange Information Requirements (EIR) are then formulated to specify the standards and guidelines for data and information exchange among stakeholders engaged in facility maintenance. EIR helps the FM department assign roles and responsibilities to provide relevant data in the required time horizon and establish communication channels and implement feedback mechanisms to facilitate effective collaboration. Moreover, EIR delineates the formats, standards, and procedures for data exchange, ensuring seamless compatibility and interoperability across diverse systems. They also establish protocols for data security, access control, and compliance with relevant regulations. Once planned, relevant information is delivered by accessing BIM models, asset management systems, and other data sources. After that, the accuracy, completeness, and timeliness of the gathered information is checked, and it is used to leverage this information to facilitate maintenance execution. Finally, from the maintenance outcomes, feedback, and experiences, missing and inefficient data requirements are defined to continuously improve the information requirement cycle for future maintenance activities, thus fostering iterative learning and optimization within the BIM-based maintenance process.

While defining information requirements for facility maintenance, literature studies and 12 CMMS tools are investigated. The information requirements reported in each tool are reported in Appendix B. These requirements are categorized in six groups: (i) descriptive information identifies asset specific information, (ii) spatial information covers spatial relations of assets and spaces, (iii) warranty information details guarantee conditions and supplier information, (iv) operational information gives brief information about operational conditions of the asset, (v) documentation reports documentative maintenance information and the remaining ones is collected.

Descriptive information of an asset encompasses crucial details that provide a comprehensive understanding of its characteristics and significance within a facility's infrastructure (Ensafi et al., 2022; R. Liu & Issa, 2016; Thabet & Lucas, 2017b). Each asset is uniquely identified by an asset ID, facilitating efficient tracking and management. The asset's name and description offer additional context, while details like make, model, and serial number delineate its specifications and origin (Y. Wang et al., 2013; Z. Wang et al., 2015). Categorization by category, family, and type aids in organizing assets for maintenance planning and resource allocation. Tags and criticality ratings further prioritize assets based on their importance to operations (Florez & Afsari, 2018; Kasprzak et al., 2013). Financial data such as purchase price and maintenance costs provide insights into investment and ongoing expenditure (Y.-C. Lin et al., 2016).

Spatial information is essential for understanding the physical layout and relationships within a facility (Ensafi et al., 2022; R. Liu & Issa, 2016). It includes precise coordinates for location tracking and building identification. Position, level, and elevation detail vertical placement, aiding navigation. Space/room designations categorize areas, while system and space served identify function and dependencies (Ensafi et al., 2022). Connections and linked assets describe relationships between spaces and assets. Maintenance accessibility paths ensure ease of upkeep (Halmetoja, 2019; R. Liu & Issa, 2013a). Spatial relations like alignment, adjacency, and connectivity illustrate how elements interact (Florez & Afsari, 2018; Halmetoja & Lepkova, 2022).

Warranty information in facility management includes details such as the manufacturer's name and contact information, providing a direct line of communication for warranty claims and inquiries (P. Dias & Ergan, 2016; R. Liu & Issa, 2016). It also encompasses the installation date, expected lifespan, and age of the asset in years, aiding in warranty assessment and planning for replacements. The warranty commencement date, end date, and duration specify the period during which warranty coverage applies, along with the entity providing the warranty (P. D. R. Dias & Ergan, 2020). Warranty content outlines the scope of coverage, while exclusions highlight any conditions or circumstances not covered. Duration extendibility conditions detail requirements for extending warranty coverage if available (Borhani & Dossick, 2020). Replacement cost estimates the expense of replacing the asset outside of warranty coverage. Additionally, supplier information for subcomponents offers insight into the sources of replacement parts and materials, ensuring compatibility and quality (Sattenini et al., 2011).

Operational information in facility management encompasses various aspects crucial for effective operation and maintenance. Operational capacity defines the maximum capability of systems or assets to perform their intended functions, guiding usage and planning (Kasprzak et al., 2013). Operational thresholds establish limits or boundaries within which systems should operate to maintain efficiency and safety (Ensafi et al., 2022). Operational schedules outline planned activities, helping to coordinate maintenance and usage to minimize disruptions. Performance metrics quantify the effectiveness and efficiency of operations, enabling performance evaluation and optimization efforts. Cut sheets provide detailed specifications and technical information about equipment or systems (R. Liu & Issa, 2013a). Sensor data and calibration records offer real-time insights into performance and ensure accuracy. Field data gathered from on-site inspections or assessments informs decision-making and maintenance planning (Q. Liu & Gao, 2017). Control commands and feedback mechanisms enable remote monitoring and adjustment of systems for optimal performance(Yang & Ergan, 2017). Asset moveability/location information tracks the position and mobility of assets within the facility, aiding in asset management and resource allocation. Lastly, spare parts are essential for maintaining equipment and ensuring operational continuity within the facility(R. Liu & Issa, 2013b).

In facility management, monitoring the current health condition of equipment is vital for proactive maintenance. A priority schedule categorizes maintenance tasks based on urgency and importance, ensuring critical issues are addressed promptly (Ali et al., 2021). Fault classes classify types of failures, aiding in diagnosis and response planning (Lucas, Bulbul, Thabet, et al., 2013b). Work orders, logs, and historical records document maintenance activities, providing insights for future planning and analysis (Lucas, Bulbul, & Thabet, 2013a; Wanigarathna et al., 2019; Yang & Ergan, 2017). Downtime for each fault class measures the duration of disruptions, guiding efforts to minimize downtime and optimize reliability. Maintenance frequency schedules routine inspections and servicing to prevent failures and maintain performance (Farghaly et al., 2017). Equipment performance metrics assess efficiency, reliability, and effectiveness, informing maintenance priorities and improvement initiatives (Sadeghi et al., 2018).

Documentation in facility management plays a critical role in ensuring the effective operation, safety, and maintenance of facilities (S. Kim et al., 2020; R. Liu & Issa, 2016; Meadati et al., 2011; Wan Siti Hajar et al., 2022; Z. Wang et al., 2015). This documentation includes manuals providing guidance on quality and maintenance repair instructions, as well as conditional assessments for evaluating equipment and infrastructure (Gu et al., 2014; Korpela et al., 2015). The code of practice for building inspection reports establishes standardized procedures for conducting inspections and reporting findings (Ali et al., 2021). Specifications outline the requirements and characteristics of equipment and materials, ensuring compatibility and compliance with standards (Kensek, 2015). Installation manuals offer step-by-step instructions for proper installation, while drawings and layouts provide visual representations of facility infrastructure and systems (Cavka et al., 2017a). Certificates attest to compliance with regulations or standards, instilling confidence in the reliability and safety of assets. Code requirements outline legal and regulatory obligations for building design, construction, and operation. As-is plans document the current state of facilities, aiding in planning and decision-making (R. Liu & Issa, 2013b). Emergency operations plans detail procedures and protocols for responding to emergencies, ensuring the safety of occupants and minimizing damage(Mayo & Issa, 2016). Safety and disaster planning documents outline strategies and measures to mitigate risks and ensure preparedness for potential hazards (Cavka et al., 2017a; Patacas et al., 2016). Instructions for training equip personnel with the knowledge and skills necessary to safely operate and maintain facilities. Environmental standards guide efforts to minimize environmental impact and promote sustainability (Lucas, Bulbul, Thabet, et al., 2013b). Maintenance and inspection reports document findings and actions taken during routine maintenance and inspections, facilitating ongoing monitoring and management (Q. Liu & Gao, 2017). Maintenance checklists provide structured guidance for conducting maintenance tasks, ensuring thoroughness and consistency (Y.-C. Lin et al., 2016). Collectively, this documentation supports informed decision-making, compliance with regulations, and the efficient and safe operation of facilities.

In a BIM environment, information for facility maintenance should be gathered and integrated into the digital model throughout the project lifecycle, starting from the early design stage and continuing up to the handover stage. Initially, asset identification and maintenance requirements are established by FM department to guarantee that the maintenance needs of the assets and their interaction are comprehensively recognized and documented. As the design progresses, this framework investigates the potential impact of geometric information and design-related issues to detect the irregularities before it is constructed. With the help of information technologies interacting with BIM environment, maintenance accessibility, regulatory compliance checks and design of the critical infrastructure that influences both facility health and occupant's comfort are analyzed virtually. While a disabled person should access to any location of the facility, in maintenance accessibility, additional issues such physical barrier needs removal (Cavka et al., 2013) or installing ladders or scaffolding, inadequate lighting or ventilation, inadequate space (R. Liu & Issa, 2013a) to perform the maintenance task, path inaccessibility to remove larger-size assets and safety issues are addressed initially (Akanmu et al., 2018, 2020; R. Liu & Issa, 2014). Moreover, some layout problems supported by clash detection of the regulatory compliance or not, are examined to avoid potential faults in the future triggering serious faults such as water damage on the structural and electrical assets and inadequate design of HVAC. In the absence of early detection during the design phase, these problems typically emerge during post-occupancy evaluations, leading to expensive retrofits and disruptions in facility operations (Seghezzi et al., 2020). Furthermore, in the design phase, detailed information about assets is refined and integrated into the evolving BIM model, including most of the spatial and descriptive information of the assets, maintenance plans and schedules. During construction, the model is updated to reflect as-built conditions and commissioning data. At the handover stage, the BIM model is finalized with comprehensive maintenance information, including manuals and schedules, and facility staff are trained in its use. By collecting and integrating maintenance information within the BIM environment throughout the project lifecycle, stakeholders benefit from a digital representation that supports efficient and effective maintenance operations, ultimately enhancing the long-term performance of the facility.

In BIM-based facility maintenance management, interoperability ensures smooth communication between different systems and stakeholders. BIM serves as the central hub, connecting with CMMS, IT solutions, and third-party tools within a Common Data Environment. This setup allows all relevant data to be stored, managed, and shared collaboratively throughout the lifecycle in a coordination of the maintenance team. Within this environment, various data exchange standards are utilized to ensure interoperability and seamless communication between different systems and stakeholders.

The Industry Foundation Classes (International Organization for Standardization, 2018) schema serves as a foundational data exchange standard in BIM. It provides a common language for representing building information across different software platforms and disciplines. For facility maintenance, IFC is augmented with the IFC-FM (Facility Management) extension, which includes specific data fields relevant to maintenance activities, such as asset information, maintenance schedules, and lifecycle data. Another key standard is the Construction Operations Building information exchange. COBie (East, 2011) defines a standardized format for organizing and exchanging facility asset data in a structured spreadsheet-like format. It captures essential information about building components, equipment, and systems, making it easier for facility managers to track and manage assets throughout their lifecycle. In addition to these standards, visual programming tools that interact with BIM environment, manipulate, and interpret BIM data to meet the specific purposes are offered as alternative for repetitive tasks with generalizable customized templates.

When collecting information needed from different vendors, OmniClass serves as a standardized classification system that ensures consistency and clarity (Thabet & Lucas, 2017a). By using OmniClass codes, information is categorized and organized uniformly, enabling easy comparison and integration of data from various sources. This standardization simplifies communication with vendors, as they can understand and provide the requested information based on common classifications. Ultimately, OmniClass streamlines the process of collecting and managing information from different vendors, enhancing efficiency, and reducing errors.

In this study, structural maturity of the information represents the easiness to access the information with different format of data and information. BIM-based information structural maturity embodies a progressive evolution from static and fragmented data sources to dynamic and integrated systems. Initially reliant on scanned documents, maintenance processes are hindered by the inefficiencies of manually searching for relevant information. The transition to searchable documentation improves accessibility and traceability, reduces the time spent searching through documents manually, and accelerates troubleshooting processes, but still lacks the cohesive organization necessary for streamlined maintenance operations. With the advancement to searchable structural information within the BIM model, maintenance capabilities were significantly enhanced through centralized, standardized, and organized data representation. The maintenance team can navigate through the structured data more efficiently, quickly locating relevant information without the need for extensive searching or manual sorting. Moreover, structured information facilitates the integration of BIM with maintenance management systems to automate workflows, analyze historical records of the maintenance data to identify trends, patterns, and recurring issues over time, and implement predictive maintenance strategies by providing insights into the condition, performance, and lifecycle of building assets to address the potential problems before they escalate (Pinti et al., 2018). Finally, the pinnacle of this evolution is realized with the integration of AI-powered chatbots for information retrieval, offering intuitive and conversational access to BIM-based maintenance data. These chatbots leverage natural language processing to provide rapid responses to maintenance queries, enabling personnel to make informed decisions swiftly and efficiently, thus optimizing facility performance and minimizing downtime. This framework strongly advocates the utilization of structural information and creating a basement for maintenance chatbot to facilitate information access accurately.

2.5.2 Information technologies in model-based facility maintenance

Information technologies are instrumental in the optimization of operational processes and the enhancement of efficiency within facility maintenance practices. This assortment encompasses a variety of sophisticated tools and methodologies such as the Internet of Things (IoT), Augmented Reality (AR), Virtual Reality (VR), Mixed Reality (MR), Geographic Information Systems (GIS), QR codes, Radio-Frequency Identification (RFID), robotics, chatbots, digital twin and blockchain. These sophisticated IT-driven solutions significantly elevate the efficacy of facility management by facilitating predictive maintenance strategies, enhancing asset tracking capabilities, and optimizing overall operational quality.

IoT:

The integration of IoT technologies with BIM-enabled facility maintenance management marks a significant advancement in building maintenance practices (Dahanayake & Sumanarathna, 2022). This integration combines BIM's detailed visualization with IoT's real-time data collection, enabling facility managers to transition from reactive to proactive maintenance approaches. It enhances real-time data collection and monitoring capabilities, allowing facility managers to monitor various building components' operational status in real-time (W. Chen et al., 2019). Sensors and smart devices collect data such as temperature, humidity, and structural integrity, feeding it back to the BIM model. This capability empowers managers to visualize the current condition of their buildings intricately (Natephra & Motamedi,

2019) , facilitating prompt responses to emerging issues and minimizing both downtime and repair expenditures. Furthermore, model-integrated IoT enables predictive maintenance through data analytics (J. C. P. Cheng et al., 2020; Villa et al., 2022). IoT sensors can detect potential issues before they escalate (Villa et al., 2021) and give alerts to warn the maintenance team. This allows for precise maintenance responses and scheduling (Hosamo, Svennevig, et al., 2022), optimizing building system performance and extending its lifespan, resulting in significant cost savings.

The quality of data collected from BIM-enabled IoT systems relies on various factors inherent to the integration of BIM and IoT technologies (Banerjee & Nayaka, 2022). Sensor characteristics such as accuracy, reliability, and compatibility with BIM infrastructure play a pivotal role in determining data quality. Moreover, the efficiency of the IoT architecture, including its support for multi-sensor integration, network protocols, and data processing methodologies tailored for BIM environments, significantly influences the reliability and accuracy of collected data. Project-specific requirements, operator proficiency, and supportive supervision also contribute to data quality. Additionally, considerations such as data storage, integration, security measures, and interoperability constraints within the BIM-enabled IoT ecosystem are critical factors in ensuring high-quality data for informed decision-making and operational optimization.

AR/VR/MR:

BIM provides a comprehensive digital representation of a facility, including geometric data, spatial relationships, and maintenance information about assets, systems, and spaces, and interacting with maintenance documentation. This enriched dataset serves as a foundation for AR, VR, and MR applications (Chung et al., 2018). As AR enhances real-world views with digital overlays, VR creates entirely simulated environments, and MR combines real and virtual elements for interactive experiences (Alizadehsalehi et al., 2020), BIM integration provides the necessary information for creating immersive experiences and visual comprehension and tracking of the maintenance issues (Natephra & Motamedi, 2019). First, incorporating a game engine (Chou et al., 2016) into this ecosystem allows the maintenance team to be trained in virtual environments by simulating various maintenance scenarios, equipment failures, and emergencies that enables them to practice their skills in a risk-free setting before they perform the tasks in the field. Hence, these simulations offer them the opportunity to thoroughly understand navigation pathways within facilities (Finco et al., 2023), access points to equipment and systems (Khalek et al., 2019), and the complexities of maintenance procedures and safety protocols (Y.-J. Chen et al., 2020). By immersing themselves in these simulations, they can encounter and troubleshoot potential problems, such as equipment malfunctions or safety hazards, in a controlled setting to refine their skills, build confidence, and ultimately enhance their ability to perform tasks safely and effectively when they transition to real-world field scenarios. Second, these solutions empower facility management teams to assess the accessibility of maintainable assets (Khalek et al., 2019) and the feasibility of maintenance routes, allowing them to practice maintenance tasks during the operational stage. This capability aids in identifying potential layout problems during the design stage, enabling necessary updates to enhance maintenance efficiency throughout the facility's lifecycle. Third, in practice, visual maintenance guides are created for the maintenance team using AR glasses or mobile devices to view digital annotations (Diao & Shih, 2019; Xie et al., 2020), spatial analysis (Feng et al., 2019), instructions (Koch et al., 2012; Mamaghani & Noorzai, 2023), work sequence (Song et al., 2020), checklists (Corneli et al., 2019), safety risk assessments and warnings (T. K. Wang & Piao, 2019) and diagrams superimposed on the physical equipment they are servicing, guiding them interactively through maintenance procedures step by step. Fourth, these tasks are supported by remote assistance with the contribution of BIM-enabled AR and MR solutions (El Ammari & Hammad, 2019). Hence, remote experts can annotate live video feeds of maintenance activities with instructions, diagrams, and annotations (Irizarry et al., 2014). This allows them to engage with the on-site team collaboratively, facilitating the troubleshooting of issues and the execution of complex maintenance tasks more efficiently through knowledge sharing and collective brainstorming.

GIS:

In facility maintenance, GIS leverages spatial data to manage, analyze, and visualize facility assets and infrastructure. Using GIS maintenance team maps the location of assets (T. Y. Lin et al., 2018; Ma et al., 2020), identify spatial relationships (Chou et al., 2016), and plan the optimal maintenance route (Hu et al., 2018), assess accessibility, and pinpoint the areas more prone to maintenance issues. By integrating GIS with BIM, the maintenance team access information about asset specifications, maintenance history, and spatial relationships directly within the GIS platform (R. Liu & Issa, 2012) and visualize spatial data to improve comprehension (Vach et al., 2018) and accurately analyze asset conditions, identify maintenance needs, plan preventive measures, and make informed decisions regarding maintenance activities (X. Wang & Xie, 2022). Additionally, BIM enhances the interoperability between GIS and other maintenance management systems, allowing for seamless data exchange and integration with workflows (Slongo et al., 2022).

RFID and QR Codes:

In practice, assets are labeled with RFID tags containing unique identification code and RFID readers are deployed throughout the facility to capture data as assets move within the space. This data is then utilized to track the assets, manage the inventory and maintenance schedules, and monitor regulatory compliances. Within BIM integration, collected RFID data is accessed in the BIM environment and linked to specifications, maintenance history, and spatial relationships (Cong et al., 2010; Sun et al., 2021). Hence, maintenance teams can visualize asset locations, movement patterns (Motamedi et al., 2013a), and maintenance status within the BIM model, improving spatial awareness and facilitating informed decision-making (Chan et al., 2016; Motamedi et al., 2013b). Moreover, BIM integrates RFID data with maintenance workflows that enables automated responses or notifications triggered by RFID events within the facility maintenance management, such as generating work orders for maintenance tasks, alerting maintenance teams about equipment failures or upcoming maintenance schedules, updating asset records with maintenance status, or triggering notifications to maintenance office about critical events or issues detected by RFID data (Kameli et al., 2021).

Similar to RFID applications, QR codes are assigned to assets, equipment, or maintenance procedures to hyperlink the physical component with digital information. QR code readers then scan them to access relevant information, instructions, or documentation associated with the QR code, facilitating efficient maintenance operations (N.-H. Pan & Chen, 2020). BIM enhances the use of QR codes in facility maintenance by integrating them with comprehensive digital models of the facility (R. Chen et al., 2015). QR codes within the BIM environment can be linked to detailed asset specifications, maintenance schedules, repair manuals, and other pertinent data (T. Y. Lin et al., 2018; Meadati & Irizarry, 2015). This integration streamlines maintenance workflows by providing instant access to critical information, improving accuracy, and enabling proactive maintenance planning. Moreover, BIM's visualization capabilities allow the maintenance team to identify asset locations, track maintenance activities, and optimize maintenance routes, enhancing overall maintenance efficiency and effectiveness (Y. C. Lin & Su, 2014). In addition to providing a direct link for accessing information and visualization, automated retrieval of relevant asset data from BIM models within the coding streamlines maintenance processes. This automation populates structural details in maintenance forms, facilitating scheduling and generating work orders for both service requests and on-site responses. This reduces on-site time and enhances the accuracy of information retrieval, optimizing maintenance operations (Chang et al., 2013). Moreover, BIM integration enhances inventory management in facility maintenance by enabling real-time updates to inventory records, synchronization with ERP systems, and tracking inventory items within the digital model (M. Wang et al., 2020).

Robotics:

Deploying robotics for facility maintenance not only involves routine maintenance tasks but also addresses risky conditions where human intervention is hazardous or impractical. Robots equipped with specialized sensors and capabilities can access confined spaces, work at heights, or handle hazardous materials without endangering human lives(Camus & Moubarak, 2015). By leveraging Building Information Modeling (BIM), robots can utilize accurate digital representations of the facility to plan and execute maintenance tasks in these risky conditions more safely and efficiently (Oyediran et al., 2021). BIM enables robots to analyze potential risks, identify obstacles, and navigate complex environments while accessing real-time data on asset conditions and maintenance requirements (Follini et al., 2021). This integration minimizes human exposure to dangerous situations, reduces the likelihood of accidents or injuries, and enhances overall maintenance effectiveness in challenging environments.

Chatbots:

Deploying AI systems to interact with users through text or voice interfaces, chatbots handle routine queries, schedule maintenance activities, provide guidance on maintenance procedures, and even troubleshoot issues remotely. With BIM integration, comprehensive digital representation and repository of the facility's assets, systems, and spatial relationships are served for training chatbots (Saka et al., 2023). Hence, chatbots can offer more personalized and context-aware assistance to maintenance personnel to streamline the maintenance workflow (K. L. Chen & Tsai, 2021). Chatbots can retrieve detailed asset information from the BIM database and provide targeted recommendations for maintenance tasks based on asset conditions, usage patterns, and historical data. Furthermore, they visualize facility layouts, asset configurations, and maintenance procedures within the digital model (K. L. Chen & Tsai, 2021). This visualization capability allows chatbots to guide maintenance personnel through complex tasks, provide step-by-step instructions, and highlight relevant information directly within the virtual environment.

Digital Twin:

Digital twins enable a new paradigm in facility management by allowing for a real-time cyber-physical system integration that supports intelligent decision-making and predictive maintenance strategies (Halmetoja, 2022). These strategies hinge on the continuous collection of data via IoT devices, followed by the application of artificial intelligence and machine learning algorithms to predict potential faults and prescribe maintenance activities (Hosamo et al., 2023; Hosamo, Svennevig, et al., 2022). This leads to cost-effective maintenance approaches, minimizes downtime, and extends the lifecycle of assets. Moreover, the integration of BIM with digital twins facilitates a more holistic approach to FM by providing detailed digital representations of physical assets, which can be updated and manipulated in real-time to reflect changes, perform simulations, and improve overall asset management (Hosamo, Imran, et al., 2022).

Blockchain:

In facility maintenance, blockchain technology offers a secure and transparent way to track maintenance activities, ensuring data integrity and accountability. It is more applicable in facility maintenance, particularly in industries where data security and transparency are crucial, such as healthcare, transportation, and manufacturing. By using blockchain, maintenance records can be securely stored and accessed, ensuring the integrity of information related to equipment maintenance, repairs, and inspections. This technology provides a tamper-proof and auditable record of maintenance activities, reducing the risk of errors or fraudulent claims (Moretti et al., 2021). Integrating Building Information Modeling (BIM) enhances this process by creating a detailed digital representation of the facility. BIM allows for better visualization and understanding of the facility's infrastructure, enabling more accurate planning and execution of maintenance tasks (T. Zhang et al., 2023). Together, blockchain and BIM improve the efficiency, reliability, and compliance of facility maintenance processes in various industries.

2.5.3 Model-based fault management

2.5.3.1 Fault detection and reasoning

Fault detection and diagnosis (FDD) methods are systematic approaches used to identify and analyze malfunctions or abnormalities in facilities. While monitoring facility operations, FDD algorithms detect deviations from the expected behavior of an asset or a system. Once a fault is detected, further analysis is conducted to determine the root cause of the deviation and formulate appropriate corrective actions. Historical records are investigated, observed deviations are correlated with other system parameters, and domain knowledge is leveraged to pinpoint the specific component or subsystem within the asset or system that is responsible for the fault. After that, based on the findings of the analysis, appropriate measures such as repairs, adjustments, or procedural changes are implemented to rectify the fault and restore the asset or system to its normal operational state.

Applying FDD algorithms in a BIM-based environment facilitates fault detection and reasoning process. These algorithms are typically employed in three distinct manners: quantitative model-based, knowledge-based, and data-driven approaches. In quantitative model-based FDDs, detailed engineering calculations or their simplified versions, relying on assumptions and approximations are employed to clarify the behavior of physical systems (Thumati et al., 2011). In this case, a digital twin of the physical system is developed within a virtual environment. This twin is employed to monitor the system and elucidate its behavior through simulations, where the calculations are embedded in the background (Hosamo, Nielsen, et al., 2022). First, the current BIM model is transformed to generate a geometric representation, which serves as the basis for energy and flow models (Shalabi & Turkan, 2020; Zimmermann et al., 2012). Additionally, pertinent technical and thermal data is extracted from the BIM and other data sources to finalize the simulation model (Hosamo et al., 2023; Zimmermann et al., 2011). After that the simulation is applied to comprehend system behavior and its results are calibrated to ensure that they accurately reflect the current condition of the physical system. The completed simulation results are compared with the current physical system’s behavior to detect the existence of faults from deviation in system behavior (Shalabi & Turkan, 2020). Once a fault is detected, the simulation inputs are either configured to pinpoint the cause of the fault under current physical conditions, or predefined customized simulation results are analyzed alongside the current data using rule-based or AI-driven methods to determine the root-cause analysis. While detailed simulation-based solutions offer a more accurate representation of system behavior, their computational intensity (Dong et al., 2014; R. Zhang & Hong, 2017) and occasional challenges in interoperability (Fernald et al., 2018), particularly in converting 3D models with detailed alphanumeric information into simulation models, present difficulties for practitioners. Hence, practitioners often turn to knowledge-based and data-driven FDD solutions as alternatives to overcome these challenges.

Knowledge-based methods are simple to develop and apply, easily interpretable (W. Kim & Katipamula, 2018) and constructed on the utilization of the prior knowledge relying on either a comprehensive understanding of the system's operations and causal relationships through domain expertise or historical records detailing fault cases to establish connections between past and present states (Zhao et al., 2019). BIM enhances knowledge-based methods by serving as a repository for detailed information about the system being monitored (Dibowski et al., 2016). Through BIM, practitioners can access comprehensive data about the structure, components, and operations of the system and extract the topology of a system (Chan et al., 2016; Golabchi et al., 2013), facilitating a deeper understanding of its behavior and potential faults. This rich information can then be leveraged to develop more accurate if-then-else rules and inference mechanisms for fault detection and diagnosis (Hosamo, Svennevig, et al., 2022). Additionally, BIM integrates real-time data from sensors and information from CMMS tools and other sources to construct model-based fault trees (Lucas, Bulbul, & Thabet, 2013b; Motamedi et al., 2014), employ failure mode and effects analysis (Lucas, Bulbul, & Thabet, 2013b), hierarchical rules that checks each fault type one-by-one (Alavi & Forcada, 2022) or generalized rules to assess the performance of the certain assets and systems such as air handling unit performance assessment rules (Hosamo, Svennevig, et al., 2022). In practice, the static rule information and dynamic incoming data are retrieved from BIM model or its interacting environment to apply the rules. Moreover, within the integration of work order records to the BIM environment, while defining new cases, previously encountered cases and their resolution are assessed via similarities between current conditions and past cases, facilitating the retrieval of relevant cases based on similarity criteria (Guerrero et al., 2022). With access to relevant cases, practitioners can adapt solutions used in similar situations to address the current fault effectively. BIM's detailed insights into the building's layout, systems, and components further support the adaptation of solutions to fit the specific context of the current situation (Motawa & Almarshad, 2013). As new fault cases are resolved, the outcomes can be stored in the BIM database, contributing to the continuous improvement of the CBR system over time. Similarly, work orders are also utilized to construct fault patterns manually. By employing problem categories, spatial and asset or system scope, and temporal fault patterns along with their associated causes, practitioners filter these causes based on matching reported problem details (Yang & Ergan, 2016a). After that, they apply specific rules to pinpoint and isolate the root cause of the issue.

Knowledge-based methods, while advantageous in their simplicity and interpretability, may encounter drawbacks within a BIM-driven environment. One limitation lies in the reliance on static rule sets derived from prior knowledge or historical data, which may not adequately capture the dynamic and evolving nature of building systems (Shu-Hsien Liao, 2005). Additionally, the effectiveness of these methods can be hindered by the complexity and scale of BIM data, which may require significant effort to extract and process. Moreover, the manual construction of fault patterns and the reliance on predefined rules may overlook nuanced or emerging fault scenarios, limiting the adaptability and accuracy of fault detection and diagnosis. These challenges underscore the need for continuous refinement and updating of knowledge-based methods to effectively leverage the rich information provided by BIM.

Data-driven FDD methods harness the power of vast datasets and advanced analytics to enhance fault detection accuracy and adaptability. Unlike model-based or knowledge-based approaches, they rely on algorithms to automatically learn fault patterns from historical data without explicitly predefined rules or assumptions. In addition to its own repository, BIM interacts with real-time sensor data, maintenance records, and other sources of information to construct comprehensive datasets that capture the dynamic behavior of building systems (Valinejadshoubi et al., 2022). Hence, the datasets are provided as inputs to train AI models. After that, machine learning techniques are then applied to analyze these datasets for the conditional statement of the assets, identify deviations signaling potential faults or abnormalities in system operations and make predictions for their future states (J. C. P. Cheng et al., 2020),. This predictive capability enables proactive measures to optimize maintenance workflows, ensuring timely interventions to prevent potential faults. Moreover, BIM supports simulation and validation of data-driven FDD methods by providing a virtual environment for testing different fault scenarios. These methods lie in their ability to adapt to changing conditions and detect previously unseen fault patterns, thus enhancing the overall effectiveness of fault detection and diagnosis in BIM-driven environments. However, challenges such as data quality, feature selection, and model interpretability need to be carefully addressed to ensure the reliability and usability of data-driven FDD solutions.

2.5.3.2 Work order management

Work order management is the systematic process of creating, prioritizing, assigning, tracking, and completing tasks related to maintenance, repairs, or other activities within a facility. Initially, work orders are generated based on various factors such as asset faults, preventive maintenance schedules, or requests from occupants (Artan et al., 2022). While creating work orders, BIM enhances step by providing a centralized digital platform that integrates comprehensive data about facility and structural information (Pinti et al., 2018). Moreover, the utilization of QR codes to access the BIM environment and retrieve pertinent data further facilitates the workflow, particularly on-site. By simply scanning QR codes placed on assets, personnel can swiftly access the associated BIM data, enabling them to create work orders efficiently and accurately (T. Y. Lin et al., 2018). Once an asset is identified as faulty, other related entries are automatically retrieved. This automation reduces manual data entry errors and ensures that all relevant and up-to-date information is included in the work order (Lavy & Saxena, 2015). Moreover, the relationships of the selected asset within the BIM environment are linked to the work order, providing valuable context for data analytics. However, if the model-based work order system is poorly designed, with complexities in navigating detailed data, integration challenges, training requirements, technical issues, and potential overload of unnecessary information, processing times may extend beyond those of traditional work order solutions (Lavy et al., 2019).

In a facility, numerous work orders are collected daily, and some of these work orders need to be prioritized based on various factors to ensure efficient allocation of resources and timely resolution of critical issues (Ensafi et al., 2023). When scheduling maintenance tasks, the availability of maintenance personnel and necessary inventories becomes the primary bottleneck determinants. The timing of tasks relies heavily on the presence of skilled staff and required materials to ensure efficient resource utilization and effective task completion. BIM improves this by integrating data on staff availability and inventory levels, allowing facilities to schedule tasks more effectively based on resource availability. Additionally, prioritization considers the criticality of maintenance work, determining its impact on overall facility operation and functionality (Kamal et al., 2021). Regulatory compliance, which legally mandates specific maintenance activities to uphold facility safety and integrity, also holds precedence. By linking maintenance tasks to relevant data within the BIM model, such as asset criticality ratings and regulatory mandates, facilities can prioritize tasks accordingly. Moreover, tasks directly affecting essential operations or posing significant safety risks are prioritized to maintain uninterrupted facility operation. Furthermore, assessment considers maintenance cost, task interdependency, urgency of the work from the point of occupants, frequency of fault occurrences, and ease of fault inspection to prioritize tasks effectively (Y. C. Lin & Su, 2014). Storing the relevant information of these factors in BIM environment facilitates model-based work order scheduling and prioritization processes and linking the spatial data with the relevant information and other technologies, the schedule of the maintenance tasks can be optimized with integrated work order prioritization, space utilization and path planning by analyzing spatial relations between assets and spaces (W. Chen et al., 2018).

Following the prioritization efforts, the planned work orders are assigned to the relevant maintenance personnel or contractor for execution. Throughout the lifecycle of a work order, it is essential to track its progress, monitor costs, and ensure timely completion. BIM facilitates progress tracking by providing real-time updates on work orders within the facility model. Maintenance team can access the BIM model to monitor the status of ongoing tasks, identify any delays or issues, and adjust schedules accordingly. Moreover, this real-time visibility enables better coordination among team members and ensures that everyone is aligned with project timelines and priorities (W. Chen et al., 2018).

BIM interacts with CMMS tools to maintain comprehensive historical logs of maintenance activities and work orders. These logs contain detailed records of past maintenance tasks, repairs, inspections, and equipment replacements. When needed, these queries are utilized to retrieve specific maintenance information (Y. C. Lin & Su, 2014). Access to this historical data enables maintenance team to track asset performance over time, identify recurring issues, and make informed decisions about future maintenance strategies. Moreover, these logs are linked to BIM assets and textual analytics are applied to work order descriptions to categorize the work orders or construct networks to comprehend them about the asset and its spatial relations from work order descriptions (Bouabdallaoui et al., 2020; McArthur et al., 2018; Nojedehi et al., 2022). However, compared to linking work orders to BIM assets, generating them in BIM environment provides structured information that makes it easier to follow the workflow.

BIM enriches the comprehension of work orders through visual representation by offering traceability to spatial and asset-specific conditional status (Firas & Yelda, 2017; K. Kim et al., 2018) alongside statistics from historical records (McArthur et al., 2018; Nojedehi et al., 2021, 2022) , and work order prioritization (El Ammari & Hammad, 2019) with color coding(Kamal et al., 2021). Furthermore, additional annotations defined within the BIM model enhance the contextual understanding of the maintenance tasks.

2.6 Discussion and conclusion

This study introduces a conceptual framework aimed at enhancing facility maintenance and fault management through BIM. By examining previous studies and practices, it covers the entire maintenance process including monitoring the facility, handling information, managing work orders, and detecting and reasoning faults. Emphasizing information management in a model-based environment, the framework addresses information challenges in facility maintenance management by leveraging information technologies to optimize maintenance operations. Therefore, this framework offers multifaceted solutions to encountered challenges. It effectively addresses various issues such as lack of information, data overload, outdated data, complex workflows, decentralized information collection, lack of standardization, poor data quality, personnel turnover problems, reporting issues, and data analytics enhancement. First, BIM addresses the lack of information by providing a centralized repository for storing and accessing comprehensive data about a facility's assets and systems. Utilizing BIM, FM department can efficiently manage and organize information starting from design phase, ensuring that essential data, such as asset specifications, maintenance history, and performance criteria, are readily available to support decision-making and optimize maintenance activities. Second, the systematic capturing and management of information, achieved through clear definition of maintenance workflow information needs, standardized data collection and management procedures, and advocacy for structured information, significantly mitigates the risk of data overload. Third, BIM ensures that maintenance activities are based on the latest information by maintaining accurate and up-to-date records of facility modifications. However, architectural, and mechanical modifications may relocate assets to different areas, remove existing assets, or replace them with alternatives that serve the same function but exhibit different behavior or features. Therefore, to preserve insights from historical work orders, information about outdated assets is stored alongside georeferenced data. Fourth, integrating standardized procedures, protocols, and guidelines into BIM enhances understanding of complex workflows and ensures consistent execution of maintenance tasks. Fifth, information provided by decentralized sources is standardized by format and content and consolidated with interoperable BIM solutions to provide efficient communication across different systems and processes. Sixth, BIM tackles poor data quality issues through rigorous data management practices and automation. By implementing quality control measures and automating data extraction processes, BIM helps maintain data accuracy and completeness. Seven, BIM-based institutional knowledge addresses personnel turnover problems. By capturing and codifying organizational knowledge within BIM, essential information is preserved and accessible even as personnel change over time. Moreover, training of the maintenance personnel in a model-driven virtual environment enhances their understanding of maintenance workflows in a risk-free, controlled setting. Eight, work orders are generated through interaction with the BIM environment. While selecting a faulty asset, asset-specific structural data entries are automatically filled in. Moreover, spatial, systemic, and parametric information of the asset is linked to the work orders. Hence, by structuring work order records and providing guided input mechanisms, BIM ensures that reports are structured, unbiased, and comprehensive. Nine, BIM-integrated data analytics enhance insights by leveraging the wealth of data stored within BIM. Through advanced analytics tools and techniques, BIM enables deeper insights into building performance, fault prediction, maintenance trends, and optimization opportunities. Finally, BIM visualization streamlines facility maintenance by providing clear, comprehensive visuals of assets and systems, enhancing understanding, communication, and collaboration among stakeholders.

BIM also enhances fault detection and reasoning by consolidating all pertinent data into a unified digital model. It enables early detection of issues during design and construction phases through clash detection algorithms and simulation tools, whereas BIM-integrated fault detection and diagnosis algorithms detect and reason the real-time faults of the assets and systems in the facility throughout the operation phase before the fault escalates. As discussed above, generating a digital twin of the facility, and analyzing its condition with a comparison of simulations is well-documented in literature but, in practice, this approach poses a time management challenge since continuous monitoring with simulations is required within a certain time intervals. Knowledge-based methods interact with BIM environment to retrieve the relevant information to apply the rules or analyze the similarities. Moreover, BIM offers an environment to link the fault with its root and influenced assets utilizing manually constructed customized templates. However, the untapped potential of BIM's intelligence and parametric capabilities is not leveraged enough to shape fault reasoning problem in the literature. Meanwhile, BIM contributes to AI-driven fault reasoning by (i) supplying data to algorithms and (ii) providing a well-organized documentation and visual comprehension of faults throughout the lifecycle of the facility. However, the success of these methods depends on the quality and relevance of the data provided by BIM and other data sources. Therefore, considerable inaccuracy in fault reasoning predictions results in less-than-optimal decisions and ineffective maintenance practices. When training the FDD algorithm using a static dataset, it may miss out on new cases, leading to ongoing mispredictions in similar scenarios. This could result in dissatisfaction among site workers, despite efforts to provide repeated feedback. Hence, continuous improvement in AI solutions is essential to adapt this dynamic nature of facility operations in practice. Besides the limited efforts for continuous improvement throughout the life cycle of the facility in the literature, collaborative works between maintenance team and AI solution need to be explored to enhance fault management.

As illustrated by red arrows in the flowchart in Figure 2.2, this framework strongly advocates for continuous improvements in BIM-integrated fault detection and reasoning. While the facility is monitored with information technologies within BIM environment, decision support solutions evaluate the collected data and information to keep the facility functional and promptly identify faults and their root causes for swift resolution. The efficiency of this support is investigated with feedback provided by the actual condition of the facility and maintenance team. Hence, decision support solutions are enhanced with this feedback consistently. Consistent with this framework, our focus is directed to BIM-integrated fault reasoning solutions. In this regard, we offer two decision support solutions to streamline maintenance workflow and enhance information management in BIM-based facility maintenance. First, we have developed a hybrid intelligence approach that facilitates human interaction with AI models to enhance fault reasoning consistently across the facility's lifecycle. This approach is tailored to meet the demands of data-driven fault reasoning while seamlessly integrating with the BIM environment to establish connections between the data-driven solution and maintenance workflows. Second, we propose a model-based work order management framework that establishes connections between assets and spaces in the BIM model and their real-world counterparts. This framework is designed to maximize information gain by leveraging the intelligence of BIM, which relies on spatial, systemic, and feature-similarity relations. It ensures comprehensive reporting of both the problem and its solution. A fault network is constructed based on observable symptoms of the fault, fault types of the root asset, and its influenced counterparts. This network is then filtered using feedback from work requesters to isolate and identify the root cause of the reported fault. Further details of proposed solutions are explored in the following chapters.

CHAPTER 3

A HYBRID INTELLIGENCE APPROACH FOR DATA-DRIVEN MODEL INTEGRATED FAULT REASONING

3.1 Introduction

Facility management is a complex discipline involving the strategic oversight of built environments, specifically focusing on facility maintenance to ensure operational efficiency and longevity. Effectively managing fault data is a crucial aspect of successful facility maintenance, allowing for the timely identification and resolution of issues, preventing disruptions, and upholding safety standards.

The incorporation of data-driven monitoring systems in facility management streamlines the process of systematically collecting, analyzing, and interpreting extensive datasets generated by diverse facility systems. Therefore, the integration of Artificial Intelligence, coupled with robust fault data management systems, empowers maintenance teams to leverage vast datasets for predictive analysis, early detection of faults, and precise diagnosis of underlying issues. This paradigm shifts towards data-driven root-cause reasoning addresses immediate concerns and facilitates proactive maintenance strategies, enhancing facility operations' reliability and efficiency.

Building Information Modeling, as a digital model, represents detailed information about the structure, assets, and systems within a facility. When coupled with data-driven fault reasoning, this digital representation serves as a dynamic and interactive platform to integrate issues into maintenance information and facilitate an understanding of system relations and flow. Maintenance professionals can leverage the rich information embedded in the BIM to identify and visualize potential faults, understand their root causes, and develop solutions.

Despite technological advancements, the significance of human intelligence in facility maintenance cannot be overstated. Skilled professionals contribute expertise and contextual understanding to complement AI-driven solutions. In human-AI interaction, the system's capabilities to interpret complex data, exercise judgment, and implement tailored solutions underscore the holistic approach necessary for effective facility maintenance management.

In this research, we investigate how the continuous interaction between humans and AI, called hybrid intelligence, improves data-driven fault detection and diagnosis to facilitate maintenance issues in a BIM-integrated environment.

3.2 Background research

3.2.1 Hybrid intelligence studies in facility maintenance

Hybrid intelligence offers a collaborative approach to bringing the strengths of human cognition with artificial intelligence together to detect, diagnose faults, and resolve issues in facility maintenance. This synergistic interaction leverages the unique capabilities of both humans and machines, fostering a symbiotic relationship where human intuition, creativity, and contextual understanding complement the computational power and efficiency of AI algorithms (Kamar, 2016). In a hybrid intelligence system, humans and machines work together to tackle the issues, each contributing their distinct advantages. This integration enhances decision-making processes, problem-solving, and overall system performance by harnessing human intelligence's cognitive diversity and adaptability alongside AI's speed, accuracy, and data-processing capabilities.

Due to the absence of existing studies on closed-loop fault management in facility maintenance, our focus is directed toward examining the maintenance workflow. We aim to explore potential human-in-the-loop and machine-in-the-loop studies to dissect and understand the specific roles of humans and machines in these loops through the maintenance workflow and operational efficiency. In practice, AI serves a dual function in human loops (Dellermann, Ebel, et al., 2019). First, it automates tasks that machines can independently handle, such as gathering operational data in a cyber-physical system, notifying the maintenance team of system faults, guiding them through diagnostic processes (Tehrani et al., 2019), resolving issues, and establishing maintenance schedules based on predefined constraints and priorities (Dellermann, Calma, et al., 2019). Second, AI provides decision support by leveraging predictive capabilities. This involves utilizing fault detection and diagnosis algorithms in fault management and assisting the maintenance team in decision-making to pinpoint the fault. On the other hand, human involvement in the machine loop enhances the capability of AI solutions. First, human expertise is required to (i) detect the faults in a system and label its type in an unidentified case before training the models in supervised classification (Ravi et al., 2017) and (ii) tune the models to improve the prediction accuracy (Dellermann, Calma, et al., 2019). Second, AI models imitate human behavior, responses, and preferences to minimize complaints and construct a tradeoff between energy efficiency and thermal comfort in the controls of HVAC operations (Jirgl et al., 2018; Kane, 2018; Meimand & Jazizadeh, 2022; Zeiler et al., 2014). Additionally, it utilizes the existence of the human as design input to optimize the model (Zeiler & Labeodan, 2019). Hence, within the human contribution, while complaints are reduced, the robustness of AI models is significantly improved in the decision-making process.

3.2.2 Fault reasoning in facility maintenance

A fault arises in a facility when an estimated parameter or observable variable associated with the process deviates from an acceptable range (Himmelblau, 1978). Within this anomaly, the operational efficiency of the system or component decreases, occasionally leading to breakdown (Singh et al., 2022). Therefore, timely identification, analysis, and addressing the fault are crucial to responding to the issue quickly and accurately to minimize downtime, prevent further damage, and ensure the efficient operation of the facility.

Fault detection and diagnosis methods have been developed to enhance the reasoning behind system faults. In identifying a fault, while the changes in the system and its symptoms are analyzed to construct the cause-effect relations, fault detection and diagnosis methods evaluate the observations and sensor readings to differentiate the features and symptoms for diagnosing the fault (Shi & O’Brien, 2019). These methods detect abnormalities, isolate their locations, and pinpoint the root causes, facilitating the maintenance of the affected components.

There have been several attempts to classify FDD methods (Abid et al., 2021; Katipamula & Brambley, 2005; W. Kim & Katipamula, 2018; Shi & O’Brien, 2019; Singh et al., 2022; Zhao et al., 2019); however, a universally accepted standard classification is yet to emerge in the literature. The inferences from those studies indicate that detailed engineering calculations are required to explain the behavior of physical systems most accurately through quantitative model-based methods (Thumati et al., 2011). Despite being computationally intensive, proposing the model also demands substantial effort (R. Zhang & Hong, 2017). Since the deviations in input can substantially influence the results, it is crucial to ensure thorough and reliable data acquisition. Therefore, they are better suited to monitor the critical systems of the distinguished facilities rather than building systems (W. Kim & Katipamula, 2018). On the other hand, the simplified calculations (Zhao et al., 2013), based on assumptions and approximations to streamline the mathematical expressions, reducing computational complexity, are more practical for routine or less complex systems, where a quick and approximate understanding of the behavior is sufficient for analyzing the fault.

Knowledge-based methods require either a thorough understanding of the system's operations and causal relationships through domain expertise or the availability of historical records detailing fault cases to establish connections between past and present states (Zhao et al., 2019). These methods are straightforward to develop and apply, with the advantage of being easy to interpret in terms of relationships (W. Kim & Katipamula, 2018). Most of these methods use a priori knowledge to derive a set of if-then-else rules and an inference mechanism exploring the rule space to identify fault symptoms for drawing conclusions on the fault. The expert system formulates rules based on prior experiences, whereas first-principle model-based approaches utilize a priori knowledge to construct a model evaluating differences between expected and actual states of the physical system. This is followed by a limit and alarm analysis to determine if analyzed objects adhere to predefined boundaries. Failure Modes, Effects, and Criticality Analysis systematically evaluates potential system failure modes, prioritizing them based on criticality for effective risk mitigation (Renjith et al., 2018). Meanwhile, Fault Tree Analysis graphically represents logical relationships between events to identify combinations leading to specific undesired outcomes, aiding in comprehending and preventing system failures (Konstantinou et al., 2011; Motamedi et al., 2014). Case-based reasoning resolves new problems by recalling and adapting solutions from similar past cases (Guerrero et al., 2022). In formulating these methods, the construction of relationships is facilitated by incorporating semantic, spatial, and temporal information (Delgoshaei et al., 2022). On the other hand, these methods are system-specific; as the system's complexity increases, the capabilities of these methods are reduced. As the inferred rules are limited to historical cases and the knowledge of the domain experts (Shu-Hsien Liao, 2005), it is challenging to identify new fault cases beyond established boundaries. Therefore, adapting to dynamic system conditions and recognizing novel cases complicates the simplicity of existing methods, as integrating new rules to accommodate unique circumstances introduces potential complexities.

In contrast to the methods mentioned earlier relying on prior knowledge on the system behavior, data-driven methods leverage the power of available data to construct the relationship between system inputs and their impact on the system behavior (Mirnaghi & Haghighat, 2020). While statistical methods examine historical data, correlating them to infer and understand the impact of system inputs on system faults, machine learning approaches model the problem as a classification task. They analyze data to capture fault patterns, especially those not explicitly defined or understood. In black box modeling, Principal Component Analysis (S. Wang & Xiao, 2004), Artificial Neural Networks (Guo et al., 2017; Hou et al., 2006; Zhou et al., 2009), Support Vector Machines (Liang & Du, 2007; K. Yan et al., 2014) and Decision Trees (R. Yan, Ma, Zhao, et al., 2016) have been proven high efficient to reason the faults (Isermann, 2006; Venkatasubramanian et al., 2003). By leveraging the information encapsulated in the data, data-driven methods can build models that capture the complex and dynamic nature of the system, allowing for the identification of patterns associated with normal and faulty behavior. This adaptability makes data-driven approaches particularly valuable in situations where a comprehensive understanding of the underlying system is challenging. However, it is essential to recognize that the success of these methods is contingent on the quality, relevance, and representativeness of the available data to exhibit the whole system behavior. Therefore, throughout the system's life cycle, relying on a constant training dataset to train the model presents certain shortcomings: (i) it fails to account for unforeseen faulty conditions as they arise, thus not adequately represented in the classification model; (ii) the classification model's limited capacity leads to repetition of the similar mispredictions during operations, consequently diminishing its practical reliability; and (iii) extrapolating beyond this data might result in notable inaccuracies (R. Yan, Ma, Kokogiannakis, et al., 2016). On the other hand, adding new instances in a living environment to the training dataset brings a significant computation burden (Thinh et al., 2019) to the model and sometimes results in challenges in clearly distinguishing the faults. Especially when the system exhibits more complex nonlinear behavior and a significantly increased number of potential fault types, the capability of the model is restrained. Therefore, careful selection and tuning of appropriate classification models are essential to ensure accurate and reliable results (Malhotra, 2015). However, even with these efforts, predicting fault classes robustly might not always be achievable. This highlights a gap in the data-driven monitoring of systems in a living environment to effectively detect, diagnose, and manage faults throughout their life cycles. Addressing this gap is vital for consistently and continuously improving fault reasoning and promptly responding to maintain operational efficiency.

3.3 Research methodology

3.3.1 Motivation

This study is motivated by the imperative need for an improved environment in the lifecycle fault management of facilities. With the rapid integration of technology and complex systems across industries, there is a growing demand for efficient fault detection and reasoning approaches that can adapt to evolving challenges. The absence of comprehensive studies on fault-based hybrid intelligence represents a significant gap in current knowledge, hindering the development of tailored solutions for the intricate fault management requirements of diverse operational environments. This research aims to fill this void by exploring and integrating a hybrid intelligence approach, combining artificial intelligence and human capabilities, to develop a dynamic fault management solution to accurately facilitate the detection and reasoning of the faults for taking precautions efficiently to keep facilities operational throughout their lifecycle.

A systematic approach to hierarchical fault problems is crucial, as complex systems often exhibit faults at different levels of hierarchy. The need for a structured fault methodology impedes effectively identifying and resolving these issues. Additionally, the study responds to the demand for a simplified classification approach to the evaluation of multiple faults successively and simultaneously. Current methods might struggle with handling numerous faults concurrently, leading to increased missing faults and fault alarms in fault detection and diagnosis of the complex system. As a result, the site workers question the reliability of the methods, as they cannot promptly identify faults that may lead to more severe problems.

Alternatively, they may invest time inspecting cases where no actual issues exist. This raises concerns about the effectiveness of the methods employed in fault detection and resolution, impacting the overall trust in the system. This research seeks to provide a systematic and simplified classification approach to handle the drawbacks of multi-class classifiers, employing a novel approach to streamline the evaluation of multiple faults, enhancing the overall efficiency of fault reasoning for facilities. Furthermore, integrating BIM into fault detection and diagnosis enables the utilization of comprehensive details regarding the building's components, systems, and their interconnections as input for enhancing design and optimizing maintenance workflows. This is achieved by offering well-organized documentation and a visual understanding of faults throughout the facility's lifecycle. By addressing these academic and practical gaps, this research contributes to developing a more resilient and adaptive fault reasoning approach tailored to complex operational environments.

3.3.2 Fault classification hierarchy

A good understanding of how to manage faults effectively is crucial for ensuring the reliable and efficient operation of a system that consists of multiple assets in a facility. Since faults in a system may arise from diverse reasons, managing and distinguishing these faults can be formidable. Consequently, to achieve successful fault management, a hierarchical approach is critical. First, it captures inherent relationships and dependencies in the hierarchy. Second, this approach reduces the complexity of the problem by breaking it down into a series of manageable sub-problems. Hence, while the number of fault issues in a system significantly increases and each exhibits nonlinear behavior that complicates their differentiation, the hierarchical relations generate more meaningful information. Therefore, in this study, we offer a fault hierarchy approach encompassing a step-by-step procedure compatible with data-driven monitoring. This approach starts with detecting a fault, followed by localization to the source asset, diagnosing the reason for the fault, and concludes with determining its severity (Figure 3.1).

A fault hierarchy begins with fault detection, which identifies deviations from expected working patterns in the system. It entails continuous monitoring of system parameters and using data-driven techniques to detect anomalies or irregularities. After detecting a fault, the next stage is to pinpoint the asset and its location within the system as a source of the problem and isolate it from the unaffected components.

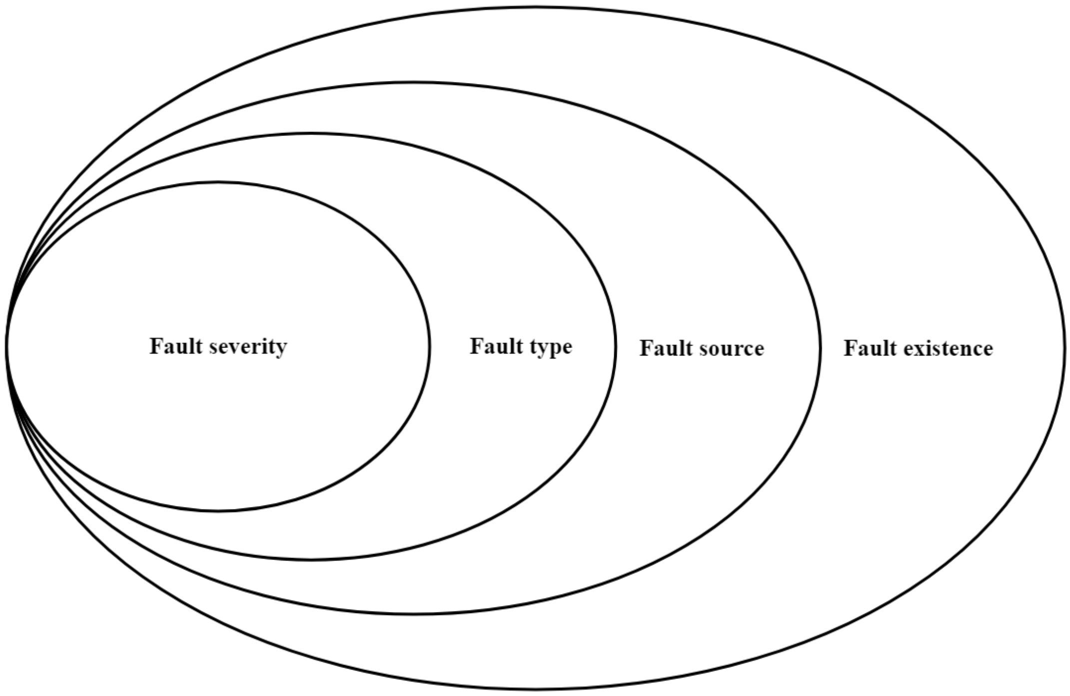


Figure 3.1. Fault hierarchy

Once the source of the fault is detected, the possible reasons that cause abnormal behavior in the system are investigated, and the specific reason is diagnosed. Finally, the intensity level of the fault is determined to assess its severity and potential consequences on the overall system. This information allows decision-makers to evaluate the condition of the fault root, enabling them to prioritize and allocate resources for timely and appropriate interventions. As a result of this proactive maintenance strategy, significant faults are promptly identified and addressed, preventing potential negative consequences or disruptions in the system.

3.3.3 A hybrid intelligence approach for fault reasoning

In this study, a hybrid intelligence approach is proposed to facilitate BIM-integrated data-driven fault reasoning in a facility. It is envisioned that humans and machines collaborate to improve decision-making continuously in analyzing the current status of a system and detecting the reason for a fault in abnormal conditions accurately as soon as possible. Hence, the on-site maintenance team promptly intervenes in identified issues as necessary, fixes the issue, and ensures that the facility functions efficiently. In this collaboration, the approach places humans in the pivotal role of decision-making in fault management, whereas the machine provides invaluable decision support through the prediction of fault reasons. On the other hand, humans play a crucial role in the system design, data preprocessing, predictive model selection, and constructive feedback to the machine predictions to augment the machine for improving the prediction accuracy.

A big picture drawn for our hybrid intelligence approach summarizes the steps followed in fault reasoning, as illustrated in its flowchart in Figure 3.2. Firstly, the facility team collaboratively works to set up a sensor network within the system. The sensor information is then integrated into a Building Information Modeling (BIM) tool, establishing a smooth connection with a BIM-enabled maintenance management system. Afterward, two different approaches can be followed to collect a large dataset consisting of both normal and faulty conditions systematically: (i) the development of an Internet of Things (IoT) system that utilizes this network to collect real-time data, or (ii) the simulation of these conditions, fine-tuned to reflect real-world scenarios as needed accurately.

Following raw data collection, facility maintenance operators carefully label the dataset, creating a hierarchical classification problem encompassing fault existence, source, type, and intensity. Subsequently, a robust training dataset is constructed by sampling it either before or after labeling, depending on the amount of data available. The data scientist in the facility team then selects an appropriate classification model for training, and efficient model settings are determined in collaboration with his

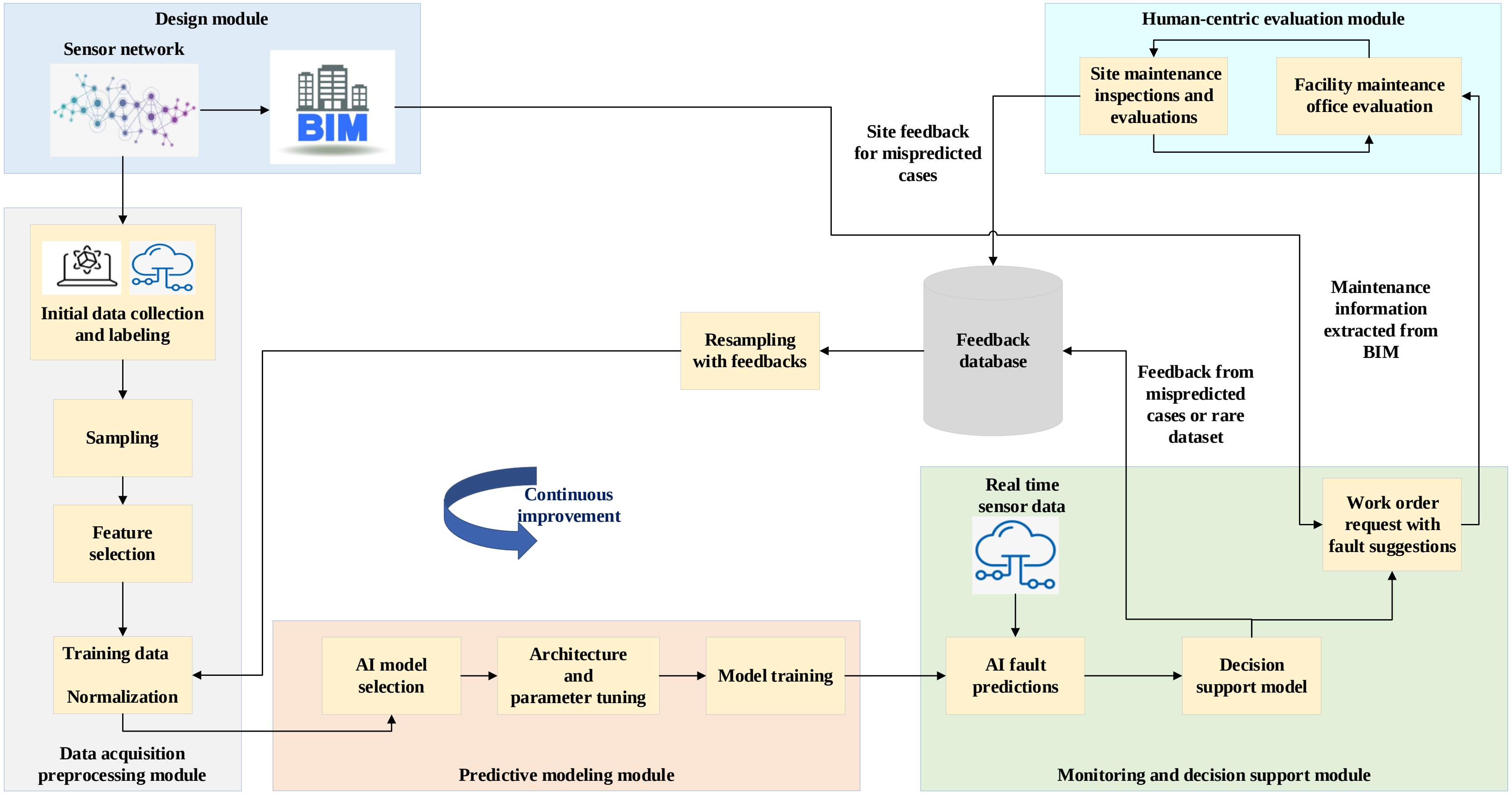


Figure 3.2. Flowchart of proposed hybrid intelligence approach for fault reasoning

guidance and the capabilities of AI. Once the model is built, it accurately predicts real-time system conditions, detecting fault existence, identifying fault sources, classifying fault types, and estimating fault intensity levels. This predictive capability significantly aids in informed decision-making for asset maintenance.

With real-time monitoring, the prediction model estimates the condition of each asset in the system at predetermined time intervals. As real-time data is collected, successive data is expected to align with either the preceding or succeeding ones.Therefore, a decision support model is deployed to follow these successive estimations and assess their consistency. As a result, it notifies the facility team for both estimations and their consistency. After that, the facility team evaluates this notification by considering the confusion matrices created from training and test data to refine the model's performance. Upon receiving the evaluation results, the site team inspects potential problem areas identified by the facility team and gives feedback to the center. Among the feedback reports, those containing mispredicted fault results and unusual inputs not present in the training dataset are stored in a database. Based on this feedback, periodic model training is then initiated, ensuring consistent updates and continuous enhancement of prediction accuracy. This establishes a closed-loop system for improved fault management.

The hybrid intelligence approach comprises six modules: design, data acquisition and preprocessing, predictive modeling, monitoring and decision support, human-centric evaluation, and continuous enhancement. The content of each module, the roles of humans and machines, and their interaction in each one are explained in detail below.

Design module:

A system comprises multiple assets, and abnormalities in each asset may result in various faults. These faults are differentiated using different types of sensors in a data-driven environment. To begin with, the facility maintenance team makes a map of the system assets and identifies those that are critical, vulnerable, and prone to faults. Following that, the team analyzes potential faults that the system may encounter. Different faults may exhibit unique patterns or characteristics specific types of sensors can capture. Therefore, the team chooses sensors aligned with fault characteristics and places them to ensure maximum coverage and sensitivity considering the proximity to critical components, environmental conditions, and potential sources of interference. As a result, the maintenance team designs the network of the sensors on the system assets, determining the most efficient placement and types of sensors for intelligent fault reasoning.

Secondly, BIM serves as an integrated platform for organizing and managing maintenance information. It consolidates maintenance manuals, task lists, and procedures within a digital representation of the facility, allowing for systematic organization and efficient planning of maintenance tasks. The visualization features aid in spatial understanding, while integration with asset management systems ensures accurate tracking of assets and maintenance histories. BIM facilitates effective communication and collaboration among stakeholders, providing a comprehensive solution for streamlined facility maintenance. Therefore, integrating sensors with BIM provides clear benefits compared to using sensors alone. When real-time sensor data is seamlessly included in the model, it offers a complete understanding of how the system in the facility is functioning. This integration allows for detailed insights about maintenance issues, as the sensor data is analyzed within the overall digital representation of the facility. Therefore, in this approach, BIM is integrated with the sensor network to (i) interact the sensors with the associated assets for extracting spatial and maintenance information and (ii) provide information about the fault assets detected to create work order request (explained in the monitoring and decision support module) to facilitate the works of the maintenance team for their inspection and maintenance tasks. This not only improves the identification of issues on site and the development of solutions but also enhances maintenance planning by utilizing the organized structure of maintenance information within the BIM model.

Data acquisition and preprocessing module:

Data is pivotal in AI, serving as the training ground for machine learning models to learn patterns and relationships for accurate predictions. More specifically, a model's abilities to generalize, avoid bias, and adapt to diverse scenarios directly depend on the dataset's quality, diversity, and representativeness (Clemmensen & Kjærsgaard, 2022).

In our approach, various faults exist in the operation of multiple system assets, which makes it necessary to collect and update the dataset carefully to train the model. Hence, we can collect the data in two different ways to adequately represent both normal and fault conditions thoroughly:

(i) Real-time monitoring data:

In this approach, the constructed IoT platform monitors the system's condition via sensor readings. It utilizes actuators to govern physical processes based on the information gathered by the sensors. For instance, if a sensor detects a high temperature, an actuator might be programmed to activate a cooling system; however, it cannot directly grasp what the cause of the problem is. It is pretty challenging to determine the faulty conditions during the monitoring for collecting its data so that it can be accomplished in three different ways. First, if such patterns are available, predefined patterns can be applied to differentiate the fault from both normal conditions and other fault types. However, this approach may be limited by the complexity and variability of fault scenarios. Second, an experienced maintenance worker may use their expertise to manually detect and label fault types during data collection. This method is effective when relying on human intuition and experience, but it is subject to individual interpretation and may not cover all possible fault scenarios. Third, the maintenance team can intentionally impose faults to collect relevant data, providing valuable insights into specific fault types; however, this approach may not be feasible for all fault scenarios and can be resource-intensive. Therefore, as the machine cannot directly analyze the fault conditional statement of the system, human involvement is required in each collection to label the fault type of the collected data.

(ii) Simulation data:

For particular cases, collecting representative and labeled actual monitoring data sufficiently and covering the variety of all possible fault types might take a considerable amount of time in the operational phase to prepare structured data for training an AI algorithm. In such cases, leveraging simulation data for model training emerges as a practical solution, offering several advantages over relying solely on real-world data. Firstly, it provides a cost-effective alternative, enabling a wide range of scenarios and environments for diverse operational settings at a lower expense than collecting extensive real-world data. Additionally, simulations allow for exploring rare faults that might be challenging to encounter in real-time monitoring. This is crucial for training models to handle unusual situations comprehensively. Its controlled nature also reduces noise and anomalies, focusing the model's learning on essential aspects. The simulation further addresses the safety concerns for risky faults where manual fault imposition might pose significant threats to the functionality of the assets in the system. Moreover, simulated data simplifies the labeling process, as precise ground truth can be easily assigned to generated scenarios. On the other hand, although simulated data provides a powerful tool for initial model development, it is crucial to incorporate real-world data in the later stages of training to ensure the model's adaptability and performance in practical applications. The synergy of simulated and real data enhances the robustness and reliability of machine learning models.

As explained above, it is essential for the data gathered to train AI models to be both representative and unbiased. This is crucial to enable the models to produce accurate and reliable results. Moreover, in such cases, since the sensors collect abundant data during the monitoring, it should be sampled with representative random input data to (i) improve the understanding and generalization capability of a machine learning algorithm with sufficient data and (ii) reduce computational expenses and time consumption (Petersen et al., 2005). In real-time monitoring, where the faults are not deliberately imposed but rather emerge naturally, data instances are first labeled and then sampled since their fault condition is unknown during data collection. Conversely, when the fault condition is known, such as in fault imposition during real-time monitoring and simulation-based analyses, the data is generated with the awareness of these conditions. For both real-time monitoring and simulation, systematic sampling that selects data points at regular intervals can be employed to capture a continuous and representative flow of data over time, facilitating effective monitoring of the system's b Similarly, stratified sampling divides the dataset into strata based on the distribution of the target classes. Each stratum corresponds to a specific fault class, and samples are then randomly selected from each stratum. This ensures that the training set maintains a proportional representation of each fault class, preventing the model from being biased towards the majority fault class and ensuring that it adequately learns from instances of all fault classes.

In addition to sampling, the collected data undergoes essential preprocessing steps. First, missing values are handled through replacement with representative values or removal. Second, duplicates and noisy data are eliminated to maintain dataset integrity. Third, outliers, which potentially distort analysis, are detected and addressed. Fourth, the input data is normalized to bring all features to the same scale, promoting consistency. Finally, input features that provide constant information but lack meaningful insights are removed. This comprehensive preprocessing approach ensures a refined and standardized dataset to enhance data quality and interpretability. All these processes are followed based on human expertise and machine capability.

Predictive modeling module:

Following data preprocessing, the next step is to train an AI model to detect and predict faults based on sensor readings. The selection of an appropriate AI model is a critical aspect of this module and holds paramount importance in determining the success of our solution for fault detection and diagnosis. In this selection, data scientists and domain (maintenance) experts collaborate to align the model with the specific requirements and nuances of fault reasoning. This collaboration is imperative since domain expertise is crucial in understanding the relationships between sensor data and fault outcomes. Hence, it can be utilized to differentiate and isolate fault types using these relations, guide feature engineering, ensure the model captures complex patterns, and evaluate AI model prediction output. Guided by feedback from domain experts, the data scientists assess various criteria to choose the right AI model: size and distribution of data, interpretability, scalability, and fitness of algorithmic structure to the nature of the problem:

- **Data size:** The amount of collected training data should be enough for the selected model to generalize well.
- **Data distribution:** The chosen model should clearly define the system's behavior to differentiate the fault types. While a linearly separable model can effectively distinguish classes using a straight line or plane, a nonlinear model is required to define the complex relationships that a simple linear decision boundary cannot capture.
- **Interpretability:** The model's predictions should be as interpretable as possible so that the facility maintenance team can gain insights into how it makes its predictions and what features or factors influence them.
- **Scalability:** The AI model should be adaptable to handle increasing training data, computational efforts, or workload without significantly degrading performance.
- **Fitness of algorithmic structure to the nature of the problem:** the chosen algorithmic approach should align well with the inherent characteristics and requirements of the problem being addressed. Given that (i) sensor reading inputs and their corresponding faults are known, (ii) each output data is categorized into fault classes, and (iii) a fault hierarchy is constructed to transform a large number of potential fault types arising in a system to be

manageable. Therefore, the algorithm's structure should be organized to solve the supervised hierarchical classification problem.

In light of the above key points, the proposed model should correspond to the nature of fault reasoning and imitate the fault hierarchy in its algorithmic structure. Therefore, it should initially check for the existence of a fault. If present, it should first predict the source asset, then identify the fault type, and finally determine its severity level. Having defined the algorithmic structure, our initial focus was directed toward understanding the operational dynamics of the system. This evaluation assesses whether the relationships between the sensor readings and fault classes are linearly separable or non-linear. Firstly, linear models were analyzed with a case study since they are relatively simple, easy to understand the model's behavior, and less susceptible to overfitting. In line with this, we employed two fundamental linear models, Logistic Regression and Support Vector Machines, for the fault existence classification problem, which represents the first step of fault hierarchy for reasoning. However, neither algorithm was able to achieve high prediction accuracy. Therefore, two common nonlinear models, Decision Tree and Backpropagated Neural Networks, were applied to the same problem to verify non-linear relationships. As expected, both models improved the accuracy of the linear models. Notably, Decision Trees produces solutions that are more prone to overfitting. Therefore, Neural Networks explain the relations more clearly with consistently accurate predictions for training and testing data. Hence, our algorithm is constructed on the basis of Neural Network classification.

Following the hierarchical structure and Neural Networks, we proposed Feedback-enhanced Hierarchical Neural Networks to predict fault reason accurately (if it exists) while monitoring the system. The details of the model are explained in the next section. While shaping the model, several vital factors significantly contribute to accurately tailoring the model to the specific nature of the problem:

- **Network chains**: The sequential arrangement of multiple neural networks in a chained architecture enables systematic progression through fault hierarchy stages. From fault existence to source asset prediction, fault type identification, and severity level determination, each network refines predictions made in preceding steps, facilitating comprehensive and accurate fault analysis. Compared to a global classifier, which is more complex and more challenging to train efficiently, with vanishing gradients during backpropagation and growing temporal dependencies, the chained networks with a top-down approach simplify the algorithmic structure and train each stage sequentially and independently with modular network architecture. Hence, it is easier to understand how information flows through the networks provided and their contribution to the final output.

- **Constructive feedback:** In the chained networks, errors can propagate through the network and affect the prediction of the following stages. For instance, if the model incorrectly identifies the source asset of a fault, it results in an inaccurate estimation of the fault type and its severity level. Therefore, to mitigate this misprediction at the final output of the entire chained network, we developed a constructive feedback mechanism to improve total prediction accuracy. Once the hierarchical model is trained, its confusion matrix, which is constructed to compare predicted and actual classifications, is analyzed to detect fault classes the model challenges to differentiate and map them to build independent relation sets. After that, assuming that additional local classifiers separate the decision boundary between the classes of each relation set efficiently in general compared to the hierarchical model, providing constructive feedback to the model. In contrast, this feedback might be misleading in some conditions, so a combined prediction of both the hierarchical model and additional local classifier is constructed to enhance the performance of the main model. Therefore, to improve the model's accuracy, we use the model outputs of both classifiers as inputs to establish a model that best fits the actual output via linear regression.

- **Randomized online learning**: The model's ability to learn and adapt for generalization is enhanced by randomizing the order of training data, ensuring a more homogeneous dataset for online learning. Because when the model is consistently exposed to the same fault in temporal data, updating its learning to accommodate new fault patterns can pose a challenge. Moreover, it is more scalable than batch learning, especially when dealing with large datasets, as models can be updated with new data as they arrive.
- **Shallow architecture:** The system's nonlinear behavior is formulated in our network using sigmoid activation functions in the hidden layer(s); therefore, to reduce the complexity of the model and make it easier to understand and interpret, a single hidden layer is utilized for contributing to a more transparent model.
- **Conditional stepwise learning schedule:** A conditional step-wise learning schedule is proposed that the model benefits from a higher learning rate in the initial stages to make more extensive updates to the weights and then decrease it when the objective (loss) function is not improved for a certain number of epochs. Additionally, the model returns to the weights calculated in the last control points to converge more slowly, avoiding divergence.

Monitoring and decision support module:

The real-time data collected by IoT devices enables continuous monitoring of the system, providing instantaneous sensor readings that serve as the foundation for predicting the fault conditions of assets. This constant prediction, however, introduces challenges related to computational resources and potential information overload. The frequent arrival of a large amount of data requires a lot of computer power to analyze it quickly. Moreover, the continuous predictions may result in overwhelming notifications for the facility team. This might make it hard to separate important issues from regular changes in the system. Therefore, a regular time interval input is defined to achieve a balance between real-time responsiveness and computational efficiency to enhance operational effectiveness.

As we collect real-time data, we expect the consecutive data points to align consistently with the data before or after them. This expectation stems from the idea that a logical and consistent flow in the system's fault condition should occur over time. Therefore, a decision support model is strategically put into action to carefully monitor and evaluate the ongoing fault predictions made by the model. As real-time data is consistently collected, this decision support model acts as a critical observer, assessing how well these predictions align with each other. Its primary role is to ensure the predictions are accurate independently and demonstrate reliable consistency over time. This consistency is examined through two essential methods: fault consistency and severity consistency.

Firstly, the model assesses fault detection consistency by analyzing sequences of prediction instances. In a case such as 'N-N-N-N-N-N-N-F-F-F-F-F-F-F-F-F-F-F-F-F,' where 'N' represents regular operation, and 'F' denotes the detection of a fault, the model is aware that a fault has been identified while the system is functioning normally, and it persists in its operation. In cases where inconsistencies arise, as seen in 'N-N-N-F-F-N-N-F-N-F-N-N-F-N-N-N-N-N-N,' the model recognizes the instances where fault detection deviates from the expected logical flow. This examination aids the decision support model in pinpointing areas where the fault detection process may need refinement. Secondly, the model examines inconsistency in fault prediction, as exemplified by the sequence 'N-N-N-F1-F1-F1-F2-F1-F2-F1-F1-F2-F1-F1-F1-F1" where 'F1' and 'F2' represent different fault types. In this case, the prediction model detects the fault accurately; however, the fault types contradict the conditional flow of the system as it changes multiple times in a sequence. Finally, despite the model accurately predicting the existence and type of faults successively, the severity level it indicates may conflict with real-time expectations. In other words, as evidenced by 'N-N-Fs1-Fs1-Fs1-Fs2-Fs2-Fs3-Fs3-Fs1-Fs1-Fs3-Fs3.' The 'Fs1,' 'Fs2,' and 'Fs3' denote increasing severity levels, respectively; in most fault cases, the asset condition in which the fault occurred is expected to deteriorate without any maintenance intervention. When the predicted severity level of the fault turns to reverse, the model discerns and notifies it.

The decision support model sequentially evaluates predictions, assesses their coherence, and ultimately decides to alert the facility maintenance team about the system's status if any abnormalities are detected. Upon achieving consistent predictions, the model interacts with a Building Information Modeling (BIM) tool to retrieve relevant maintenance information. Then, it utilizes this information to create a work order request for the facility maintenance team. On the other hand, for the contradictory predictions, the model stores both sensor readings and prediction outputs in a control database. Moreover, it analyzes the confusion matrix of training and test sets to determine whether this contradiction is caused by an edge condition where the model struggles to make accurate predictions. Finally, it notifies both situations (two different faults or different severity levels of the same fault or one is a fault, and the other is a normal condition) with their maintenance information extracted from the BIM model, reporting the availability of edge condition. Therefore, the assessment of the situation is entrusted to human judgment. Moreover, the model detects unusual sensor readings that are not present in the training sets and stores them in the control database to enhance the model's generalization capability, which will be updated at a later stage.

Human-centric evaluation module:

Humans are responsible for determining fault reasoning in the system, while machines (AI) provide decision support to facilitate human reasoning. Therefore, the first prediction model estimates the fault statement of the system. Then, the decision support model analyzes the consistency of the real-time predictions to generate a work order request for the faulty condition and send it to the maintenance team. Hence, this request is processed through the team's filter to finalize the insight into the reason for the fault in the system. As introduced, two different types of requests are generated in the previous module based on the consistency of the fault predictions. In the first one, the maintenance experts either confirm AI prediction for the fault condition or evaluate the given condition differently if they see a statement different from the machine estimates. After that, they assign the worker order to the site team to inspect the situation on-site. Secondly, the contradictory fault predictions need more attention to reach a decision. In this case, different solution approaches can be followed. In the first approach, when the maintenance team possesses adequate experience to conduct expert evaluations, they assess both conditions with respect to given data readings using their expertise, historical records for the reported assets, and a human understanding of the patterns embedded in the AI prediction model. At the end of the evaluation, they prioritize the inspection of the faults on site and report it to the assigned site team. While their knowledge is insufficient, they assign the work order to the relevant technical personnel on-site and direct details of the request to them. At this time, the site team quickly analyzes the faults and decides to start from which asset fault occurs. Besides human expertise, AI predictions and the criticality of the assets are also considered to prioritize the inspection of the faults if needed. After that, the site team starts the examination of the prior faulty asset and follows the required maintenance task using maintenance information provided in the BIM tool and their experience. To summarize, all the process followed in the approach aims to (i) monitor the working system, (ii) determine the root of the fault in the system as soon as possible, and (iii) reach the corresponding information via BIM to manage the maintenance flow efficiently and timely.

Continuous enhancement module:

The evaluation of the prediction model does not always accurately pinpoint fault detection and reasoning within the system. The subsequent model-based decision support and human expert re-evaluation corrects some of these mispredictions; however, in some cases, the actual fault of an asset is identified on-site during the inspection, contradicting the initial assumption of the prioritized faulty (another) asset. Therefore, the prediction model's increasing number of these misestimated cases significantly burdens the central maintenance team and amplifies the workload for the site team. Moreover, following the site inspection and maintenance task, the site team updates the work order to describe the encountered problem and its solution for keeping it in historical maintenance records. Therefore, repeating similar faults detected during inspections demotivates the site maintenance team since the feedback from previous cases goes unaddressed. This raises concerns about the reliability of the proposed prediction model.

As introduced above, a control database is constructed to store the sensor readings and their predictions periodically for contradictory cases, faulty conditions, and out-of-sample sensor readings. Using feedback reports from the site, each case is labeled. After that, the labeled and predicted outputs of the same sensor readings are compared. Among these, the mispredicted samples are filtered, and the representative ones are incorporated into the training dataset. Hence, the prediction model undergoes retraining using the updated dataset. As the various NN models are retrained in our hierarchical chain networks to facilitate the process, the finalized weight and architecture of the previously updated model are utilized as the initial point to train the model. The prediction model is updated periodically to continuously enhance the prediction accuracy in fault reasoning, thereby decreasing human effort and improving time management in maintenance works.

3.3.4 Feedback-enhanced Hierarchical Neural Networks

In this study, we developed a prediction model named Feedback-enhanced Hierarchical Neural Networks to solve hierarchical classification problems, specifically for accurately predicting faults and their reasons in the monitored system.

3.3.4.1 Hierarchical model

In our model, we applied a top-down approach to organize fault classes in a hierarchical structure and form their relationships as a rooted tree structure. To further elaborate, the foundation of this structure is rooted in the existence of a fault, serving as the primary node. Upon detecting a fault, the model initiates an investigation into the source of the fault, branching into internal nodes that represent different sources. Once the source is identified, the model proceeds to determine the specific fault type associated with that source, further refining the hierarchical structure. Finally, at the leaves of the tree, the model predicts the severity level for each identified fault type. In other words, the model is structured into four levels of hierarchy, and these levels are chained through parent-child relationships. The analysis progresses through these levels based on conditional statements, each providing more detailed information about the fault.

The model orchestrates multiple local classifiers based on Neural Networks to formulate the problem in harmony with the final prediction. It initiates the decision-making process at the root node, depicted in Figure 3.3, employing a binary classifier to determine whether a fault exists within the system. While traversing down the hierarchy, it assigns a local classifier to each parent node in the hierarchy and trains each one independently using associated data with the child classes under its corresponding parent node. The number of child nodes in this hierarchy influences how the classifier is structured. If only one child node exists, no additional classifier is necessary, as the decision is straightforward. When two child nodes exist, a binary classifier predicts between these two classes. However, if there are more than two child nodes, a multi-class classifier is applied to predict the specific child class within the set. It allows each local classifier to specialize in distinguishing between classes at its hierarchy level. This adaptive approach ensures the classifier aligns with the hierarchical complexity, enhancing the model's ability to make accurate predictions.

3.3.4.2 Proposed neural network model

A neural network is a machine-learning model inspired by the structure and function of the human brain (Wu & Feng, 2018). It consists of interconnected neurons arranged into layers and is designed to learn and recognize complex patterns and make predictions. In principle, neurons are the building blocks of the nervous system. They are capable of receiving input, processing information through weighted sums, applying activation functions, and producing outputs (B. Cheng & Titterington, 1994). Our neural networks consist of an input layer, one hidden layer,

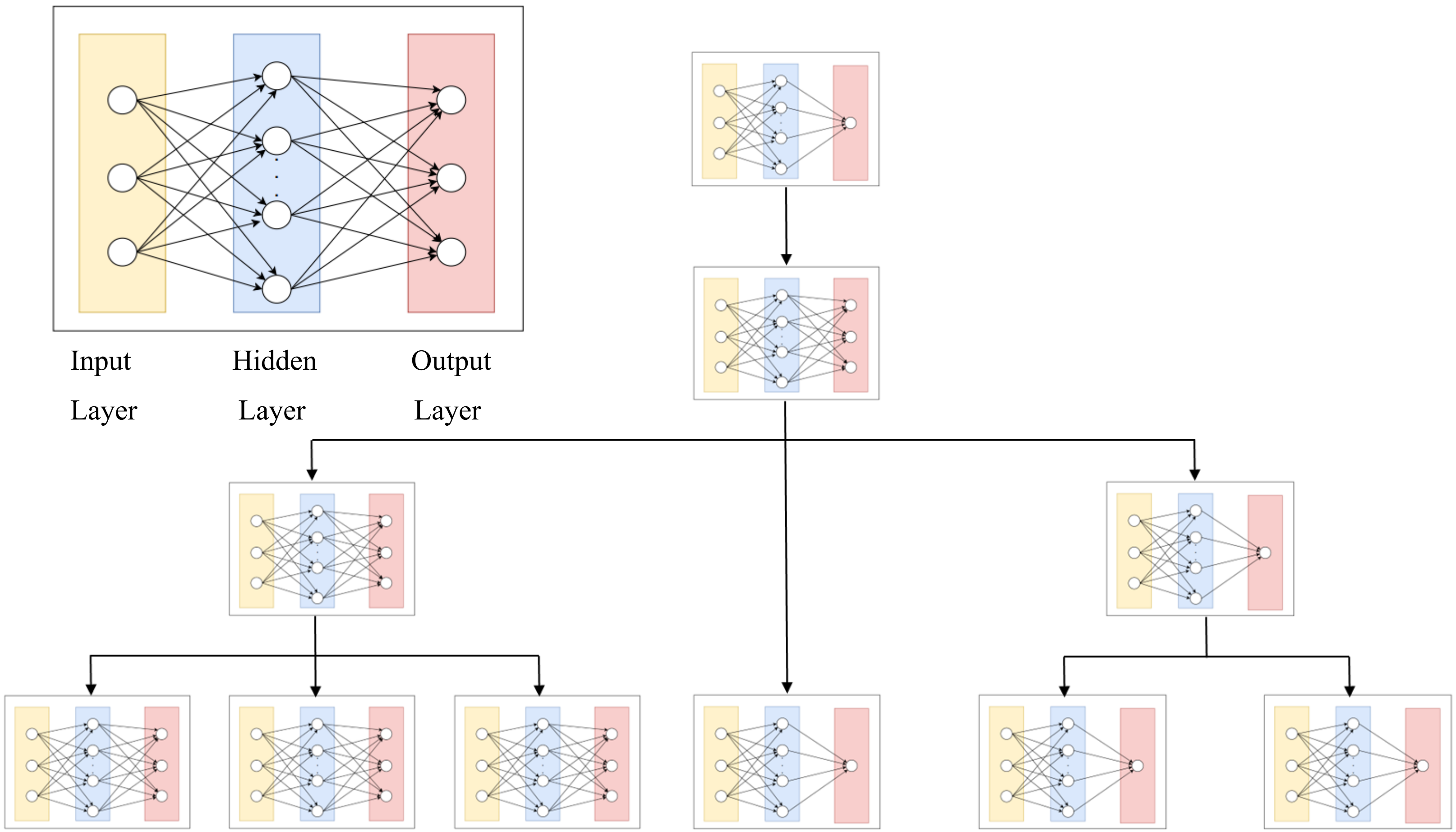


Figure 3.3. A typical example of a hierarchical classification model

and an output layer.

A backpropagation algorithm is applied to optimize the weights assigned to connections between nodes during training, adapting to the patterns in the training data and improving the network's ability to make accurate classifications(Hecht-Nielsen, 1989). The algorithm comprises two fundamental steps: forward pass and backward pass. During the forward pass, input data propagates through the network layers to generate predictions. The computed predictions are then compared to the target output using a loss function. In the backward pass, the gradient of the loss with respect to each weight is calculated by applying the chain rule of calculus. These gradients indicate the direction and magnitude of adjustments needed for each weight to minimize the loss. The weights are updated in the opposite direction of their gradients using a stochastic gradient descent algorithm and conditional staircase learning schedule. This process is iteratively repeated until one of the termination criteria is satisfied: the network converges to a state where the loss is minimized, or the prediction accuracy of the optimized network is not improved.

In our model, to improve the convergence ability of the network, we apply a randomized online learning mechanism, conditional staircase learning schedule, and sigmoid activation functions to the hidden layer. Randomizing the order of training data in online learning enhances the model's adaptability for generalization, prevents overfitting to consistent faults in temporal data, and provides scalability by allowing continuous updates with new data, which is particularly beneficial for large datasets (Pérez-Sánchez et al., 2018). Moreover, the conditional staircase learning schedule adjusts the learning rate in response to the model's performance to avoid overfitting and underfitting. While it initially uses a higher learning rate to converge faster, at specific epochs, it checks the convergence of the model. In cases where the model diverges by generating a higher accumulated loss function or lower model accuracy compared to the previous checkpoint, the model returns to the connection weight values in the last checkpoint and decreases its learning rate to improve its performance. Furthermore, we select sigmoid activation functions on hidden layers to effectively capture the non-linear behavior of the model, surpassing the efficiency of linear and rectified linear unit functions (Daqi & Yan, 2005). Additionally, for the output layer, a sigmoid function is employed in binary classification, while a softmax function is utilized in multi-class classification (Szandała, 2021). The log-loss function (in Eq. 3.1) is applied to binary and multi-class classifiers (Janocha & Czarnecki, 2017).

$$\text{ALF} = \begin{cases} \frac{1}{N_{data}} \sum_{n=1}^{N_{data}} y_{nc} \log(p_{nc}) + (1 - y_{nc} \log(1 - p_{nc}) & if\ C_{class} = 2 \\ \frac{1}{N_{data}} \sum_{n=1}^{N_{data}} \sum_{c=1}^{C_{class}} y_{nc} \log(p_{nc}) & C_{class} > 2 \end{cases} \tag{3.1}$$

where y, p, ALF, N_{data}, and Cclass present actual output, predicted output, accumulated loss function, number of training, and number of classes, respectively.

The pseudocode of our neural network model is presented in Table 3.1.

Table 3.1 Pseudocode of the Proposed Neural Network

```
Design the network: determine the classifier type (binary or multi-class), select the number of neurons in the hidden layer and activation functions in both hidden and output layers, define the staircase learning set, and set the maximum number of epochs (MaxEpoch), number of epochs to check model convergence (ModCheck), number of training data (Ndata), length of learning rate set(MaxLR), model accuracy (Acc) and accumulated loss function (ALF)

t=1; i=1; Acc_old=0;  ALF_old = Ndata;
while (t < MaxEpoch) and  (i<=MaxLR)
  for n= 1: Ndata
     Apply forward pass
     Calculate loss function
     Follow backward pass
     Update connection weights (W) using the gradient descent with the corresponding learning rate
  end
  if mod (t, ModCheck) = = 0
     Apply forward pass for all training data
```

```
    Calculate the accumulated loss function for all training data (ALF_new)
    Predict the classes
    Compare the predictions with the actual ones and determine the accuracy of the model  (Acc_new)
        if   Acc_new < Acc_old
             i=i+1
             W=W_old
        elif  ALF_new > ALF_old
             i=i+1
             W=W_old
        else
             Acc_old = Acc_new
             ALF_old = ALF_new
             W_old= W
  end
  t=t+1
end
```

3.3.4.3 Feedback mechanism

As explained above, the hierarchical model finalizes its fault reasoning prediction through a structured sequence of four hierarchical decision processes, which are independently trained. Naturally, the local classifiers in four hierarchy levels must predict the relevant classes accurately to determine the right fault reason within the system. In other words, a single misprediction in any of the four decision levels leads to a flawed final decision. Therefore, understanding how and why errors propagate throughout the hierarchy is crucial for debugging and giving feedback to improve model performance.

In order to develop a constructive feedback mechanism for our model, we initiated the analyses from the designed hierarchical model and trained local classifiers. We then investigated whether the hierarchical model encounters difficulties in distinguishing between the classes and, more specifically, the local classifier on each node struggles to separate the decision boundaries of the child classes. To accomplish this, the finalized confusion matrix compares the ground truth and predicted classes for all training data. We examined the confusion matrix, detected all sets causing striking false classifications between two classes, and mapped them. In some cases, the hierarchical model faces challenges in differentiating more than two classes. Consequently, we grouped these classes instead of employing pairs. Finally, we proposed additional local classifiers for these subsets to gain information from them and give constructive feedback to the main hierarchical model to improve model accuracy. On the other hand, in specific instances, this feedback to the main model might be deceptive; therefore, combining both additional classifiers' and the main model's predictions is more comprehensive to generate robust estimations. In this study, we employed least square regression to minimize the square error between actual faults and feedback-enhanced predictions, optimizing the combined weights (w) of the hierarchical main model and additive classifier for each subset. While the weights are optimized to address the more critical faults in two-class subsets (Eq. 3.2a), they are adjusted uniformly across all classes in multi-class subsets (Eq. 3.2b).

$$minimize\ \frac{1}{N_{data}}\sum_{n=1}^{N_{data}}\left(y_n - w_1\,p_{a,n} - w_2\,p_{m,n}\right)^2 \qquad (3.2a)$$

$$minimize\ \frac{1}{N_{data}}\sum_{n=1}^{N_{data}}\sum_{c=1}^{C_{class}}\left(y_{nc} - w_1\,p_{a,nc} - w_2\,p_{m,nc}\right)^2 \qquad (3.2b)$$

$$subject\ to \quad w_1, w_2 \geq 0 \qquad (3.3)$$

whereas p_a and p_h are the prediction rates of additional classifier and hierarchical model, respectively.

Inherent to their design, both models are anticipated to enhance the optimization of the final model through positive weights. However, in some cases, a negative weight may emerge at the conclusion of the least square regression. In such cases, if the hierarchical model bears a negative weight, the feedback from the additional classifier is directly acknowledged, and the weight is updated as w=[1 0]; otherwise, no further feedback is deemed necessary.

Moreover, to mitigate the adverse effects of error propagation within the hierarchical model, the additional classifiers are strategically designed to offer feedback to either a parent node or a network at a higher hierarchy level without incurring any loss of information, if possible. To illustrate, for the first case, the local classifier at the node corresponding to source A is designated to classify its fault types, namely A1, A2, A3, and A4; however, it encounters difficulty in distinguishing between A1 and A3. Hence, the additional classifier is trained to delineate the boundaries between A1 and A3 more distinctly, furnishing constructive feedback to the classifier at the source A node. On the other hand, when confronted with the challenge of separating normal condition and severity level 1 of the fault type A2 within the model, and in the absence of any misclassifications between the normal condition and other severity levels of A2, the additional classifier can be tailored to classify the normal condition and the A2 fault type. Similarly, this condition applies to source A if none of the fault types pertaining to source A is mispredicted in conjunction with the normal condition. However, if the model encounters confusion where an instance from another severity level is mistakenly identified as the normal condition or vice versa, it leads to information loss in the model. Therefore, currently, feedback is selectively provided only for classes with normal conditions and severity level 1 of fault type A2.

In this study, we proposed different constructive feedback mechanisms to improve the performance of the main hierarchical model as explained below:

(i) Global information gained with an additional multi-class classifier:

In such cases, the main model faces challenges in differentiating between various fault types and/or a no-fault condition. Therefore, by utilizing the training ground truth outputs and inputs for these classes, an additional multi-class classifier is trained. On the other hand, the prediction rate of these classes in the main model is calculated using Eq 3.4, multiplying the prediction rate of its corresponding class at each hierarchy level up to its level L_p.

$$p_{m,nc} = \begin{cases} \prod_{l=1}^{L_p} p_{l,nc} & if\ fault\ exists \\ 1 - p_{l=1,nc} & o/w \end{cases} \quad (3.4)$$

(ii) Global information gained with an additional binary classifier:

In this scenario, the main model encounters difficulty in distinguishing the two classes, either two different fault types for a specific severity level or one for a faulty condition, and the other is normal. Rather than involving both classes in determining weights as in multi-class cases, the binary classifier concentrates on the critical one and updates the weights of the linear regression according to this fault class. Hence, an additional binary classifier is trained using the data of both classes, referring to the instances as "1" as the critical one and "0" as the other. On the other hand, the prediction rate of the critical one for the training instances is calculated using Eq. 3.5.

$$p_{m,n} = \prod_{l=1}^{L_p} p_{l,n} \quad (3.5)$$

(iii) Local information gained for a multi-class classifier:

In contrast to the previous approaches, we directly focus on the local classifier on the corresponding parent node and remove training data of its clearly separated classes. After that, a new classifier is trained to classify the remaining classes. While the prediction rate is computed using Eq 3.6 for a multi-class classifier for providing feedback, Eq. 3.7 applies to binary classifiers where the critical fault is selected as reference among the two classes.

$$p_{m,nc} = p_{l=L_p,nc} \quad (3.6)$$

$$p_{m,n} = p_{l=L_p,n} \quad (3.7)$$

(iv) Local information gained for a binary classifier with an additional binary classifier:

In such a condition, both local and feedback classifiers utilize identical data instances to train their models. Therefore, repeating the same procedure twice does not contribute additive feedback to the main model. In such cases, either the classification algorithm can be substituted, or the architecture of the neural network is modified to enhance the learning process.

Integrating constructive feedback from additional classifiers into the main model involves optimizing the contribution weight of both models through linear regression. To ensure a systematic approach, the implementation order of various new classifiers supporting the main model is determined based on the independence of subsets and the criticality of faults. Initially, classifiers with independent subsets that do not influence the prediction of the remaining classes are applied to the model. Then, those containing critical fault(s) are implemented, followed by considering the remaining classifiers. During each implementation, the performance of the updated model is assessed using measures such as, primarily, model accuracy and, secondarily, distribution of misclassification. If the model's performance demonstrates improvement, the weights of the feedback approach are stored, and the model undergoes an upgrade. Conversely, if there is no enhancement in performance, the feedback is not accepted, and the model remains unchanged. This iterative process is followed for all supportive classifiers, ensuring that only beneficial feedback contributes to the model's refinement.

The pseudocode of the step-by-step explanation of our feedback approach is presented in Table 3.2 below:

Table 3.2. Step-by step explanation of the proposed feedback mechanism

Step 1: Set a threshold value for the number of mispredictions in the confusion matrix of the hierarchical model
Step 2: Identify two-class subsets that exceed the threshold misprediction number and construct a multi-class subset if all pair combinations are available in the subsets, eliminating those pair sets.
Step 3: Determine how to integrate the additional classifier on each subset for constructive feedback to the main model
Step 4: Train all additional classifiers using data samples with the corresponding ground truth classes
Step 5: Establish the sequential implementation order for integrating the additional classifiers with the main model. Give priority to the classifiers, including independent subsets and critical faults.
Step 6: Apply constructive feedback of each subset to the main model one by one and update the model if the model accuracy is improved.
Step 7: Follow the same procedure in the training set and use the same weights for linear regression of feedback provided by additional classifiers to predict faults for the test set. Report confusion matrix and other performance metrics.

3.4 Validation case studies

This research validates the efficiency and practicality of the hybrid intelligence approach proposed for fault reasoning through the examination of two case studies. Due to challenges in setting up experiments and the unavailability of specific data, the datasets initially created for air handling unit (AHU) fault classification (Granderson & Lin, 2019) were reorganized to align with our solution approach. Therefore, it is assumed that the sensor network was predesigned, and the design inputs were comprehensively defined and integrated with the BIM model. Hence, the case studies focus more on the efficiency of the proposed classification algorithm and decision support model and the effects of human-machine collaboration for continuous improvement.

The performance of the proposed classification algorithm is tested with these case studies and compared with commonly known classifiers:

(i) linear multi-class classifiers: Logistic Regression, Support Vector Machines,

(ii) nonlinear multi-class classifiers: Decision Trees, Feed-forward Neural Networks, Back-propagated Neural Networks

(iii) hierarchical models of Decision Trees and Back-propagated Neural Networks

Each classifier is trained and tested in MATLAB using the same datasets with input normalization on the interval [-2 2]. While the built-in functions are applied for the classifiers: Logistic Regression `fitcecoc('Learners','linear') with` $C_{class}*(C_{class}-1)/2$ binary classifiers; Support Vector Machines `fitcecoc('Learners','svm')` with linear kernel function whereas other hyperparameters such as regularization parameter, kernel scale and strategies for extending binary classifiers to handle multiple classes are optimized internally; Feed-forward Neural Networks `fitcnet()` where the weights of the network are optimized by Limited-memory Broyden-Fletcher-Goldfarb-Shanno algorithm and Decision Trees `fitctree()` whose hyperparameters including split criterion, at least one observation per child node, number of maximum splits and number of predictors to select at random for each split are optimized internally; Back-propagated Neural Networks are implemented in the MATLAB script. The architecture of each Neural Network comprises a hidden layer with 20 neurons activated by the sigmoid function, where the output layer is activated by either the softmax function for multi-class classification or the sigmoid for binary classification. The number of neurons in the hidden layer is experimentally decided. The number of neurons in the hidden layer of a neural network is typically determined empirically through experimentation and optimization.

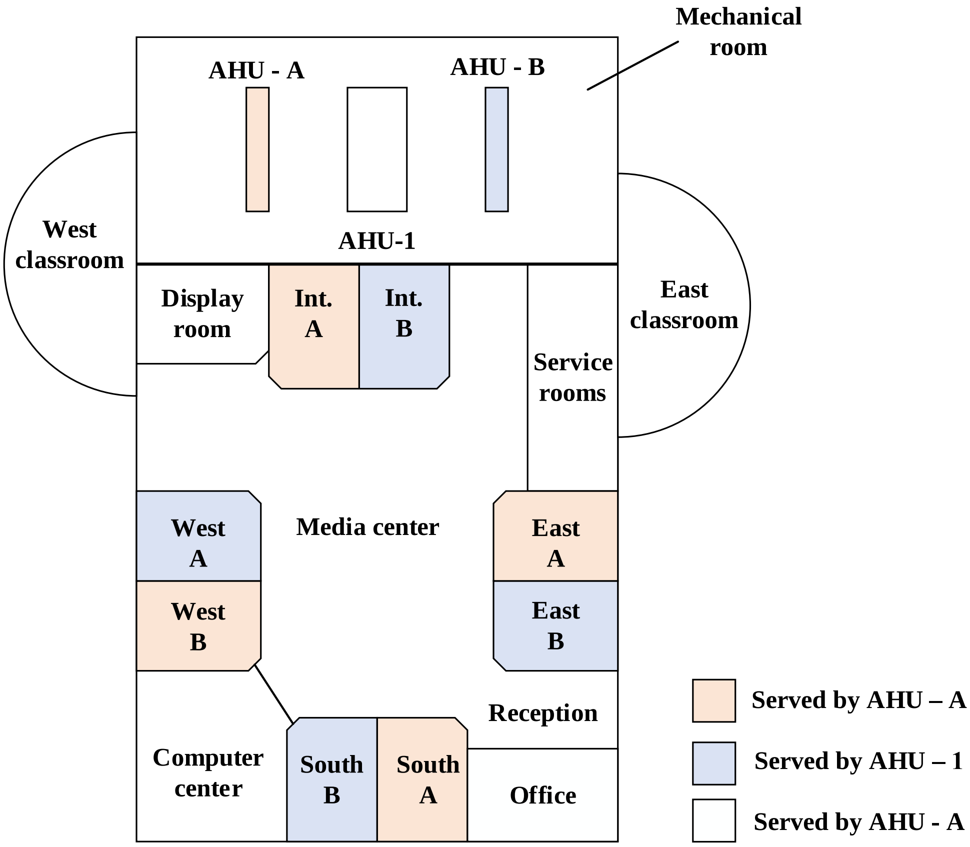
3.4.1 Case Study 1: Simulated multi-zone variable air volume air handling unit

3.4.1.1 Building and system description

In this case study, we applied a simulated fault dataset generated by Drexel University as part of the ASHRAE 1312 project (Granderson & Lin, 2019). The dataset is specifically designed for a small-scale test facility at the energy resource station in Iowa, built to facilitate the comparison of different energy efficiency measures and the documentation of energy consumption. The facility is configured for side-by-side testing, incorporating three AHUs, as shown in Figure 3.4. AHU-1 is responsible for the common areas, while the remaining AHUs are dedicated to the A- and B-Test Systems. In this case, we focus on AHU A, which serves four zones, three of which are externally exposed, and one is confined to internal conditions.

AHU-A consists of multiple major assets: the supply air and return air fans; preheat, cooling, and heating coils, as well as heating and cooling control valves; and integral elements include recirculated air (RA), exhaust air (EA), and outdoor air (OA) dampers, alongside the requisite ductwork for facilitating the conveyance of air to and from the conditioned spaces. The fans are responsible for circulating air within the system. The supply air fan pushes conditioned air into the space, while the return air fan draws air back into the unit for further conditioning. The coils play a crucial role in adjusting the temperature of the air. The preheat coil warms the air, the cooling coil removes heat, and the heating coil adds heat as needed to achieve the desired temperature. On the other hand, dampers regulate airflow to optimize ventilation and air quality, while valves control fluid flow to precisely manage the thermal characteristics of air, collectively enhancing system efficiency and indoor environmental control. To further elaborate, these dampers control the proportions of recirculated air (air already present in the building), exhaust air (air being expelled from the conditioned space), and outdoor air (fresh air from the environment) that are mixed and circulated through the system. Simultaneously, valves provide thermal

(a)



(b)

Figure 3.4. (a) 3D view and (b) floor plans of the studied energy resource station building in Ankeny, Iowa

regulation by managing the flow of hot or cold water through the heating and cooling coils, thereby controlling the temperature of the air being supplied. This integrated control of airflow and thermal conditions ensures a comprehensive approach to the system operation, optimizing both air quality and temperature control for occupant comfort and energy efficiency.

The system operates in an occupied mode from Monday to Sunday, spanning the hours of 6:00 am to 6:00 pm. During this timeframe, specific control parameters are implemented to regulate the HVAC system, ensuring optimal environmental conditions within the building. The supply fan and return fans maintain continuous operation, while the cooling coil valve modulates to uphold a fixed 55°F supply air temperature when the outdoor air damper is at a minimum and mechanical heating is required. In the mechanical heating mode, the AHU heating coil valve adjusts to maintain a fixed 65°F supply air temperature. The supply fan with variable frequency drive (VFD) and return fan operate with static pressure and speed control sequences, respectively. Minimum outdoor air control is set at a fixed 40% opening when the unit is not in economizer mode. The economizer mode is triggered below 65°F outdoor air temperature, and the OA damper and return air damper are adjusted to maintain the supply air temperature setpoint with the cooling coil valve closed.

3.4.1.2 Dataset description

In this case, the fault dataset presented by Granderson and Lin (Granderson & Lin, 2019) is employed. It was generated within a simulation environment. Therefore, HVACSIM+ software was utilized to model the dynamic behavior of the AHU-A system and four associated building zones involving four variable air volume (VAV) boxes. This modeling process was conducted for both typical operation and fault conditions across summer, winter, and transition seasons. Each fault was simulated throughout the day, and data points were recorded at 1-minute intervals. The simulations for faulty conditions mainly focus on the three main assets of the air handling unit: outdoor damper and valves of heating and cooling coils, which regulate both temperature and air supply within the system. Specifically, in the simulation, scenarios involve the damper and cooling coil valve getting stuck while the heating coil valve exhibits leakage. These faults are introduced by setting fixed control signal values for the damper and cooling coil valve to simulate being stuck and manually opening the heating coil bypass valve to simulate leakage. In the process of generating the dataset, four distinct levels of getting stuck for dampers are simulated: "fully closed," "40% open", "45% open", and "55% open." Similarly, the cooling coil valve is modeled to be stuck in positions such as "fully closed," "fully open," "partially open 15%", and "partially open 65". In contrast to the instantaneous stuck of these two components in any position, the leakage in the heating coil valve gradually increases as it deteriorates. Therefore, it is classified into three different classes: "Stage 1: 0.4 GPM", "Stage 2: 1.0 GPM," and "Stage 3: 2.0 GPM". On the other hand, a fault-free condition is simulated throughout different times of a year. Hence, to encompass all possible simulated conditions, the outputs are illustrated in Figure 3.5 using a hierarchical fault tree, organizing the hierarchical model.

Throughout the simulation of the diverse range of operational and fault conditions within the main parts of the air handling unit (AHU-A) system, external parameter settings and internal reactions governing the system’s behavior and responses were reported as design inputs to generate this dataset. These inputs, tabulated in Table 3.3, consist of air temperatures measured in outdoor air, supplied air, mixed air, and return air; control signal to adjust (i) the damper position of relevant air types, (ii) the valve position of the heating and cooling coils separately, and (iii) fan speed of both supply and return air; supply air temperature set point; and measured supply air duct static pressure.

Since only the occupancy mode is under investigation, both the supply and return air fan statuses are maintained as “on,” and the targeted duct pressure of the supply remains constant. As a result, these constant parameters are not subject to evaluation. In total, 13 distinct design inputs are assessed to construct the predictive model.

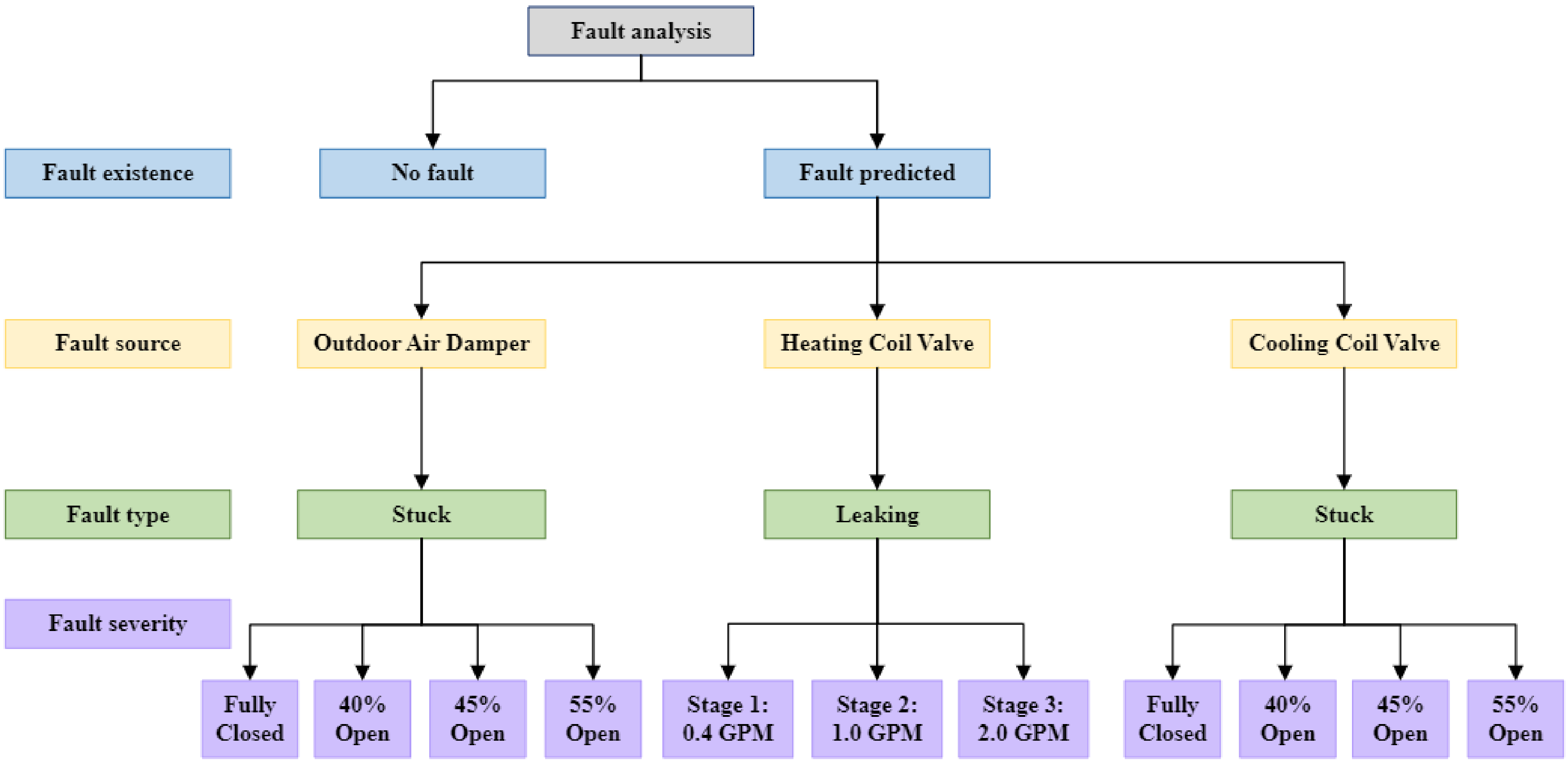


Figure 3.5. Fault hierarchy of case 1

Table 3.3 Design inputs of case 1

Input	Description
Supply Air Temperature	Measured AHU supply air temperature
Supply Air Temperature Set Point	AHU supply air temperature set point
Outdoor Air Temperature	Measured AHU outdoor air temperature
Mixed Air Temperature	Measured AHU mixed air temperature
Return Air Temperature	Measured AHU return air temperature
Supply Air Fan Speed Control Signal	AHU supply air fan speed; ranges from 0 to 1; 0 - fan speed is 0%, 1 - fan speed is 100%
Return Air Fan Speed Control Signal	AHU return air fan speed; ranges from 0 to 1; 0 - fan speed is 0%, 1 - fan speed is 100%
Exhaust Air Damper Control Signal	The control signal for the AHU exhaust air damper ranges from 0 to 1; 0 – damper should be fully closed, 1 – damper should be fully open.
Outdoor Air Damper Control Signal	The control signal for the AHU outdoor air damper ranges from 0 to 1; the 0 damper should be fully closed, and the 1 damper should be fully open.
Return Air Damper Control Signal	The control signal for the AHU return air damper ranges from 0 to 1; 0 – damper should be fully closed, 1 – damper should be fully open.
Cooling Coil Valve Control Signal	Control signal for AHU cooling coil valve ranges from 0 to 1; 0 – valve should be fully closed, 1 – valve should be fully open.
Heating Coil Valve Control Signal	Control signal for AHU heating coil valve ranges from 0 to 1; 0 – valve should be fully closed, 1 – valve should be fully open.
Supply Air Duct Static Pressure	Measured AHU supply air duct static pressure

3.4.1.3 Model description

The dataset explained in the previous section is utilized to validate the continuous improvement of our hybrid intelligence approach. In other words, the network involving the design inputs in the first module and data to train and test the model is given. Therefore, we first reorganize the dataset to fit it to the formulation nature of our hybrid approach. As detailed below, three distinct datasets are essential for

implementing the model: (i) a training dataset for the initial model, (ii) a test dataset for the initial model, and (iii) a test dataset for the updated model. Therefore, the simulated data points, arranged in a time series, are organized into five groups, each containing consecutive points at fixed five-interval intervals, as illustrated in Figure 3.6. Each group comprises 1872 data points representing "no-fault" scenarios, along with 288 points corresponding to "outdoor damper stuck in a fully closed position." Additionally, there are 144 data points for each of the remaining ten classes. All design input data is normalized to scale the values within the interval of [-2, 2].

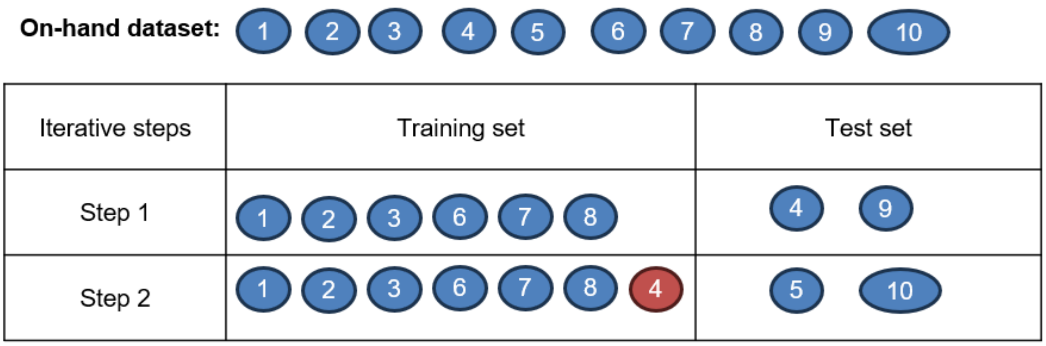


Figure 3.6. A sample representation of training and test sets in iterative steps

Two iterative steps are followed to test the hybrid approach. In the first iteration, our feedback-enhanced hierarchical model is trained with the data points from the first three groups, cross-validated with 3-folds reported in Appendix C, and then tested with the data from the fourth group. Following this, the decision support model and human evaluations are employed to assess the accuracy of the predictions. Feedback from this step is incorporated by adding selected misclassified fault samples to the training data for updating the existing model in the second step. The model is then trained with the updated dataset and tested with the data from the fifth group. Simultaneously, the previous model is tested with the same set of data points for the sake of comparison, aiming to explore potential improvements in the updated model.

In implementing the predictive model, the hierarchical model sequentially orchestrates multiple local classifiers to predict the presence of faults in AHU-A operations and determine the specific reason if a fault is detected. The following local classifiers are proposed:

(i) In the first level of the hierarchy, a binary classifier is trained to predict whether a fault exists.
(ii) In the second level, a multi-class classifier is developed to classify the source asset of the faults,
(iii) As each component has a single fault type, no classifier is assigned,
(iv) In the lowest level, three different multi-class classifiers are employed for each fault type.

Following the prediction of the trained classifiers in the sequential hierarchy, a confusion matrix is constructed for the training dataset. Based on the confusing class sets with more misclassifications than the threshold value, additional local classifiers are developed to provide feedback to the model and enhance its performance. The threshold value is set to 10.

The performance of both the initial and updated versions of the proposed model is compared with different multi-class classifiers hierarchical models introduced above.

3.4.1.4 Results

3.4.1.4.1 Performance analysis of hybrid intelligence approach

As introduced in the previous section, the efficiency of the proposed approach for fault reasoning is investigated following at least two iterative steps. In the first iteration, the hierarchical model undergoes training using multiple local classifiers, each specializing in a specific part of the fault hierarchy. In this model, the binary classifier initially predicts the existence of a fault in the system with an accuracy rate of 98.76%. Moving to the next hierarchical level, where the focus shifts to identifying specific boundaries within possible fault sources, such as the outdoor damper and the valve of heating and cooling coils, the model showcases robust performance. The multi-class classifiers effectively delineate fault boundaries, with only one misprediction observed out of 5534 data instances. This result highlights the model's precision in localizing faults within fault sources. Finally, as expected, the local classifiers at the lowest level of the hierarchy precisely classify the severity level of each fault type. As a result of this, our hierarchical model predicts the training data with 12 distinct classes with an accuracy rate of 98.75%. The predictions and ground truth fault classes are compared in a confusion matrix to show the performance of the hierarchical model. It appears that the model encounters challenges in distinguishing between two specific class pairs:

(i) "0:No-fault" vs "4-2:Stage 2: 1.00 GM level leaking of heating coil valve"
(ii) "0:No-fault" vs "4-6:Stuck cooling coil valve in a fully closed position".

Considering the threshold value, these particular pairs exhibit more misclassifications. In the first one, 42 fault-free instances are incorrectly predicted as faulty conditions, while 54 fault data instances are missed by erroneously assessing that no fault exists. These are 13 and 27 data instances in the second one. Therefore, implementing a feedback mechanism is essential to enhance the model's ability to differentiate between these classes accurately. As '2: Stage 2: 1.00 GM level leaking of heating coil valve' and '6: Stuck cooling coil valve in a fully closed position' can be effectively differentiated in the hierarchical model, two additional binary classifiers are introduced. These classifiers serve to provide valuable feedback on the main model. Since both faults are assumed to be at the same criticality level, the priority for implementing feedback is determined by the number of misclassifications observed in both pairs. Hence, the feedback from the first pair, which incurred 96 misclassifications, is initially implemented, followed by the feedback from the second pair with 40 misclassifications.

The confusion matrix reveals a significant imbalance in data distribution between 'no fault' cases and other fault conditions used for providing feedback. Additionally, the model effortlessly distinguishes 'no fault' instances from different severity levels of leaking heating and stuck cooling coil valves. Hence, utilizing fault sources instead of the specified severity level of their fault types may offer more valuable feedback to the main model, addressing issues related to balanced data distribution and information loss resulting from its hierarchical structure. The following additional classifiers are developed to gain global information and improve the capability of the main hierarchical model with constructive feedback:

(i) A binary classifier to distinguish “0:no-fault” and faults of “2:2heating coil valve”,
(ii) A binary classifier to distinguish “0:no-fault” and faults of “2:3 cooling coil valve”,

In this context, the faulty classes are assessed as 'critical' and serve as a reference in the feedback calculations. Following the training of additional classifiers, linear regression is utilized to determine the weights of the additional classifiers and the main model, facilitating the update of the overall model. In the first case, the weights are optimized as $w_{2,02}$=[1.0303 -0.0203]; however, it is corrected as $w_{2,02}$=[1 0] since the negative feedback of the main model to the overall one is not acceptable. Within this upgrade, the overall model corrects nearly half of the misclassifications in the hierarchical model. Therefore, the first feedback is accepted. In the second case, the regression model adjusts the weights as $w_{2,03}$= [0.8303 0.1691], where the additional classifier contributes more to the overall model. This update indicates that the overall model enhances accurate predictions by one instance in total, as outlined in Table 3.4; however, the missed faults decrease to 5 cases (previously 27), whereas false fault alarms increase to 34 (previously 13). From the maintenance office's perspective, an increase in false fault alarms adds to the workload of the site team, requiring inspection of reported faults and raising concerns about the predictive model's reliability. Conversely, missing faults have the potential to impact the system's functionality. Given the higher criticality of functionality in this case, the feedback from the additional classifier is deemed constructive and accepted. Consequently, this feedback enhances the accuracy rate of the model to 99.21%.

We tested the performance of the hierarchical model with a set of test instances, using the multiple trained classifiers arranged in a predefined hierarchy. The results show that the hierarchical model predicts the classes of the faults with an accuracy rate of 98.42%. However, similar to challenges faced during training, the model struggles to differentiate between pairs of the same class. In the first pair case, it incorrectly classifies 19 fault-free instances as faulty while missing 17 actual faults, resulting in misclassifications of 5 and 11 instances in the second case. As reported in the confusion matrix of the test cases in Table 3.5, introducing feedback classifiers to the overall model, incorporating optimized weights obtained during training, reduces false fault alarms to 13 instances in the first case, while keeping the number of missed faults the same. In the second case, the feedback mechanism decreases the total number of misclassified instances by three; however, missed faults decrease to 3, while false fault alarms increase to 11. Consequently, this feedback mechanism corrects nearly 17% of the misclassified instances in the hierarchical model and enhances the overall accuracy rate to 98.69%.

The decision support model assesses successive model predictions of test instances at five-minute intervals. It identifies a single-step change in prediction as a false prediction, while the model flags suspicious prediction patterns for human evaluation through work order requests. These patterns include (i) the sustained prediction of the same reason after a change in at least two successive instances (e.g., 0-6-6-0, 8-0-0-0) and (ii) predictions changing at least four times within one or two interval steps (e.g., 0-6-0-6-0-0-6, 0-8-0-0-8-0-8). Therefore, the model recognizes 19 instances out of 49 as false predictions and stores them in the database. The remaining predictions are found suspicious and reported to the maintenance office through 7 work order requests, comprising three false alarms and four missing faults. It is assumed that the site team inspects both heating and cooling coil valves and reports the results. Without any feedback, the workload of the site team could have escalated, necessitating four additional inspections, encompassing three false alarms

Table 3.4 Confusion matrix of the training dataset trained by the initial feedback-enhanced hierarchical model for case 1

Training Set		Predicted											
		No-Fault	Outdoor air damper (stuck)				Heating coil valve (leaking)			Cooling coil valve (stuck)			
			Fully closed	40% open	45% open	55% open	Stage 1: 0.4 GPM	Stage 2: 1.0 GPM	Stage 3: 2.0 GPM	Fully closed	Fully open	Partially open 15%	Partially open 65%
Actual	No-Fault	5561	1	0	0	0	0	20	0	34	0	0	0
Outdoor air damper (stuck)	Fully closed	1	863	0	0	0	0	0	0	0	0	0	0
	40% open	0	0	432	0	0	0	0	0	0	0	0	0
	45% open	0	0	0	432	0	0	0	0	0	0	0	0
	55% open	0	0	0	0	431	0	0	1	0	0	0	0
Heating coil valve (leaking)	Stage 1: 0.4 GPM	0	0	0	0	0	432	0	0	0	0	0	0
	Stage 2: 1.0 GPM	27	0	0	0	0	0	407	0	0	0	0	0
	Stage 3: 2.0 GPM	0	0	0	0	0	0	0	432	0	0	0	0
Cooling coil valve (stuck)	Fully closed	5	0	0	0	0	0	0	0	428	0	0	0
	Fully open	0	0	0	0	0	0	0	0	0	864	0	0
	Partially open 15%	0	0	0	0	0	0	0	0	0	0	432	0
	Partially open 65%	0	0	0	0	0	0	0	0	0	0	0	432

Table 3.5 Confusion matrix of the test dataset trained by the initial feedback-enhanced hierarchical model for case 1

Training Set		Model											
		No-Fault	Outdoor air damper (stuck)				Heating coil valve (leaking)			Cooling coil valve (stuck)			
			Fully closed	40% open	45% open	55% open	Stage 1: 0.4 GPM	Stage 2: 1.0 GPM	Stage 3: 2.0 GPM	Fully closed	Fully open	Partially open 15%	Partially open 65%
Actual	No-Fault	1848	0	0	0	0	0	13	0	11	0	0	0
Outdoor air damper (stuck)	Fully closed	0	288	0	0	0	0	0	0	0	0	0	0
	40% open	0	0	144	0	0	0	0	0	0	0	0	0
	45% open	0	0	0	142	1	1	0	0	0	0	0	0
	55% open	0	0	0	0	143	0	0	1	0	0	0	0
Heating coil valve (leaking)	Stage 1: 0.4 GPM	0	0	0	0	0	144	0	0	0	0	0	0
	Stage 2: 1.0 GPM	17	0	0	0	0	0	126	0	1	0	0	0
	Stage 3: 2.0 GPM	0	0	0	0	1	0	0	143	0	0	0	0
Cooling coil valve (stuck)	Fully closed	3	0	0	0	0	0	0	0	133	0	0	0
	Fully open	0	0	0	0	0	0	0	0	0	287	0	1
	Partially open 15%	1	0	0	0	0	0	0	0	0	0	143	0
	Partially open 65%	0	0	0	0	0	0	0	0	0	0	0	144

and one missed fault case. Hence, using both direct prediction of the decision support model and site feedback, 42 misclassified samples are evaluated and added to the training dataset to update the overall model.

Following the update of the training dataset, all local classifiers within the hierarchical model undergo retraining. During this training session, each previously trained classifier's final version from the preceding step serves as the starting point for updating, facilitating an accelerated learning process. Consequently, the hierarchical model is reconstructed with the updated local classifiers. Upon the inclusion of new samples in the training dataset, the prediction accuracy of the original hierarchical model experiences a decline to 98.46%. In contrast, the updated hierarchical model demonstrates an improvement in accuracy, reaching a rate of 98.60 %. The comparison of actual and predicted fault reasoning classes shows the challenges to distinguish fault-free conditions with the “Stage 2: 1.00 GM” level of heating coil leakage and stuck cooling coil in the fully closed position as reported in Table 3.6. Therefore, the same local classifiers are retrained to provide feedback to the updated model.

Table 3.6 Confusion matrix of the training dataset for selected fault classes trained by the updated feedback-enhanced hierarchical model for case 1

Training Set		Predicted		
		No-Fault	Stage 2:1.0 GPM Leaking of heating coil valve	Stuck cooling coil valve at fully closed position
Actual	No-Fault	5554	46	27
	Stage 2:1.0 GPM Leaking of heating coil valve	67	381	0
	Stuck cooling coil valve at fully closed position	14	0	430

After retraining these additional classifiers, linear regression is employed to update the contribution rates of both the additional classifier and the updated hierarchical model, aiming to enhance the overall model. In the first case, the optimized weights are determined as $w_{u,2,02}$=[0.9855, 0.0154], signifying a substantial influence of the additional classifier on the decision-making process. Within this contribution, the overall model successfully rectifies the prediction of 46 instances out of 114. In the second case, the regression model refines the weights to $w_{u,2,03}$=[0.9179, 0.0827]; however, this adjustment does not yield a noticeable enhancement in the prediction accuracy of the overall model. As a result of this, only the first feedback is accepted and integrated into the overall model. This feedback enhances the prediction accuracy of the overall model to 99.00%, as illustrated in Table 3.7, which presents the finalized confusion matrix of the training set.

Finally, the performance of the overall model is assessed with new test instances. The hierarchical model exhibits a commendable accuracy rate of 98.58%, accurately predicting most cases. However, challenges persist in distinguishing between pairs of the same class. By incorporating only constructive feedback provided from the first case, the model successfully decreases false leaking alarms to 13 instances (compared to the previous 17 in the hierarchical model), and missing leaking faults decrease to 15 (previously 21), as reported in Table 3.8. Hence, this feedback corrects ten predictions, leading to an updated prediction accuracy of 98.90%.

The decision support model tracks the predictions and identifies the patterns for the successive incoming ones. Out of 41 misclassifications, the model detects nine instances, recording them in the database. In contrast, the model recognizes some contradictory patterns for the remaining cases and prepares four work order requests to the maintenance office to evaluate the situation. These requests involve two false alarms and two missing faults. Compared to the hierarchical model, the number of human interventions decreases by two times. Additionally, the model updates contribute to a reduction in the necessity for on-site inspections.

Table 3.7 Confusion matrix of the training dataset trained by the updated feedback-enhanced hierarchical model for case 1

Training Set			Model											
			No-Fault	Outdoor air damper (stuck)				Heating coil valve (leaking)			Cooling coil valve (stuck)			
				Fully closed	40% open	45% open	55% open	Stage 1: 0.4 GPM	Stage 2: 1.0 GPM	Stage 3: 2.0 GPM	Fully closed	Fully open	Partially open 15%	Partially open 65%
Actual	No-Fault		5558	0	0	0	1	0	38	0	31	0	0	0
	Outdoor air damper (stuck)	Fully closed	2	862	0	0	0	0	0	0	0	0	0	0
		40% open	0	0	432	0	0	0	0	0	0	0	0	0
		45% open	0	0	0	432	0	0	0	0	0	0	0	0
		55% open	0	0	0	0	432	0	0	1	0	0	0	0
	Heating coil valve (leaking)	Stage 1: 0.4 GPM	0	0	0	0	0	432	0	0	0	0	0	0
		Stage 2: 1.0 GPM	30	0	0	0	0	0	418	0	0	0	0	0
		Stage 3: 2.0 GPM	0	0	0	0	0	0	0	433	0	0	0	0
	Cooling coil valve (stuck)	Fully closed	10	0	0	0	0	0	0	0	434	0	0	0
		Fully open	0	0	0	0	0	0	0	0	0	864	0	0
		Partially open 15%	0	0	0	0	0	0	0	0	0	0	432	0
		Partially open 65%	0	0	0	0	0	0	0	0	0	0	0	432

Table 3.8 Confusion matrix of the test dataset trained by the updated feedback-enhanced hierarchical model for case 1

Training Set		Model											
		No-Fault	Outdoor air damper (stuck)				Heating coil valve (leaking)			Cooling coil valve (stuck)			
			Fully closed	40% open	45% open	55% open	Stage 1: 0.4 GPM	Stage 2: 1.0 GPM	Stage 3: 2.0 GPM	Fully closed	Fully open	Partially open 15%	Partially open 65%
Actual	No-Fault	1843	0	0	0	1	0	15	0	13	0	0	0
Outdoor air damper (stuck)	Fully closed	0	288	0	0	0	0	0	0	0	0	0	0
	40% open	0	0	144	0	0	0	0	0	0	0	0	0
	45% open	0	0	0	144	0	0	0	0	0	0	0	0
	55% open	0	0	0	0	144	0	0	0	0	0	0	0
Heating coil valve (leaking)	Stage 1: 0.4 GPM	0	0	0	0	0	144	0	0	0	0	0	0
	Stage 2: 1.0 GPM	10	0	0	0	0	0	134	0	0	0	0	0
	Stage 3: 2.0 GPM	0	0	0	0	1	0	0	143	0	0	0	0
Cooling coil valve (stuck)	Fully closed	1	0	0	0	0	0	0	0	143	0	0	0
	Fully open	0	0	0	0	0	0	0	0	0	288	0	0
	Partially open 15%	0	0	0	0	0	0	0	0	0	0	144	0
	Partially open 65%	0	0	0	0	0	0	0	0	0	0	0	144

In conclusion, we introduce a hybrid intelligence approach to enhance fault reasoning predictability throughout the life cycle of a system. The effectiveness of this approach is tested by comparing the performance of the original hierarchical model with the updated feedback-enhanced hierarchical model using test instances in the second step. Results indicate that the original model achieves a prediction accuracy of 98.46%, while our model improves accuracy by accurately predicting 29% of previously misclassified instances, leading to an overall accuracy of 98.90%. This improvement comprises a 0.12% contribution from regular classifier updates, and the feedback mechanism significantly contributes with a 0.32% enhancement in total. Moreover, these improvements reduce the workload by decreasing the need for human intervention for decision-making. Therefore, this underscores the robustness of our model in addressing the hierarchical fault reasoning problem.

3.4.1.4.2 Comparative analyses of FEHNNs with AI methods

The performance of our classification algorithm, Feedback-enhanced Hierarchical Neural Networks, is tested with both initial and updated training and test datasets and compared with other commonly known classification algorithms:

(i) Logistic Regression and Support Vector Machines as linear multi-class classifiers,
(ii) Decision Trees and Neural Networks, which include both Feed-forward Neural Networks and Back-propagated Neural Networks, as nonlinear multi-class classifiers,
(iii) Hierarchical models of Decision Trees and Back-propagated Neural Networks.

The comparison results for both datasets are detailed in Tables 3.9 and 3.10. As indicated in the results, within the additional misclassified samples in the original training dataset, the newly trained classifiers exhibit a decrease in their training accuracy compared to the original ones. Overall, the algorithms display similar

Table 3.9 Prediction accuracy of different classification algorithms using original dataset for case 1

	Training Accuracy (%)	**Testing Accuracy (%)**
Logistic Regression	91.56	91.67
Support Vector Machines	91.59	91.67
Feed-forward Neural Network	96.97	96.66
Decision Tree	99.07	98.16
Hierarchical Decision Tree	98.96	97.92
Back-propagated Neural Network	98.45	97.76
Hierarchical Neural Networks	98.75	98.42
Feedback Enhanced Hierarchical Neural Networks	**99.21**	**98.69**

Table 3.10 Prediction accuracy of different classification algorithms using updated training set for case 1

	Training Accuracy (%)	**Testing Accuracy (%)**
Logistic Regression	91.36	91.64
Support Vector Machines	91.36	91.64
Feed-forward Neural Network	96.60	96.88
Decision Tree	98.94	98.56
Hierarchical Decision Tree	98.97	98.34
Back-propagated Neural Network	98.38	98.42
Hierarchical Neural Networks	98.60	98.58
Feedback Enhanced Hierarchical Neural Networks	**99.09**	**98.93**

behavior with minor differences. To begin with, linear classifiers, including Logistic Regression and Support Vector Machines, consistently achieve accuracy levels exceeding 90%, establishing a robust baseline for comparison. However, nonlinear classifiers prove to offer more resilient solutions in comparison to their linear counterparts, attaining a notable accuracy of over 96%.

In nonlinear comparison, while Decision Trees yield more accurate predictions for training sets than various versions of multi-class Neural Networks, there exists a noticeable gap between its training and testing accuracy, indicating potential overfitting. Hence, BPNN exhibits better performance than with decision trees in test sets. In the case of FEHNN, it not only enhances the performance of its root algorithms but also achieves the highest testing accuracy among all the algorithms considered.

Comparing the performance of our hierarchical neural network model with other neural network architectures, such as the Feed-forward Neural Network and Back-propagated Neural Network, reveals the superiority of the hierarchical approach. While the Feed-forward Neural Network exhibits slightly lower accuracy, the hierarchical models, especially FEHNN, demonstrate better control over-generalization. Additionally, the FEHNN model outperforms the Back-propagated Neural Network in both training and testing accuracies. In contrast, the hierarchical model based on Decision Trees does not achieve solutions with the same level of accuracy and efficiency as the multi-classifier counterpart since the hierarchical model requires mitigating overfitting to prevent the accumulation of information loss throughout the hierarchical structure. Therefore, our solution showcases its effectiveness in capturing hierarchical relationships and incorporating feedback mechanisms. These results underline the advantage of employing a hierarchical structure in neural networks, particularly with feedback enhancements. This approach enhances the generalization capabilities of the hierarchical models and leads to superior performance and robustness in addressing this fault reasoning problem.

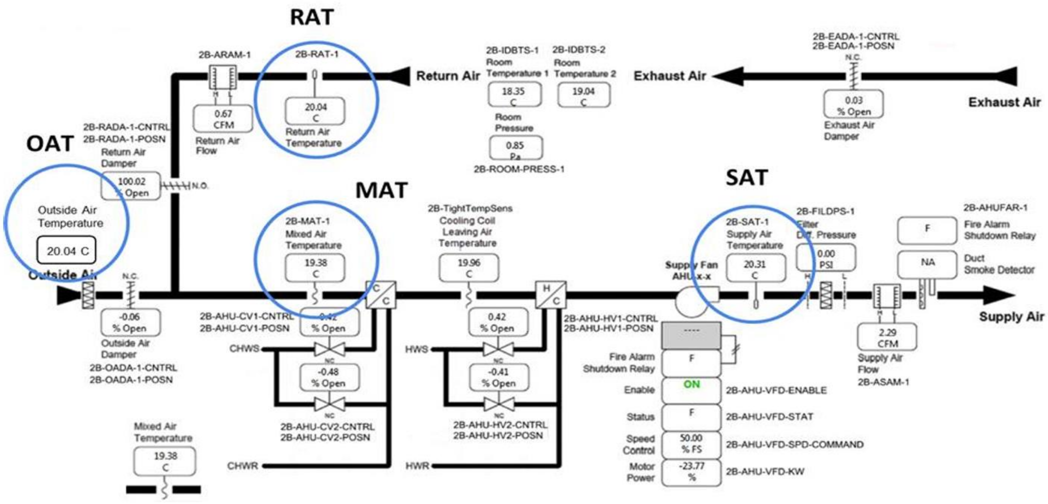
3.4.2 Experimental case study 2: experimental single-zone variable air volume air handling unit

3.4.2.1 Building and system description

In this case, we applied the experimental dataset for a Single Zone Variable Air Volume (SZVAV) Air Handling Unit generated at Lawrence Berkeley National Laboratory (LBNL) in their FLEXLAB test facility (Granderson & Lin, 2019), figured out in Figure 3.6. The AHU used in the experiments served test cell X3A and featured major components such as a supply air fan with a VFD, cooling and heating coils, cooling and heating control valves, outdoor air, return air, and exhaust air dampers where it is schematic diagram is given Figure 3.7. The functional roles of each reported asset remain consistent with those in the initial case study. Throughout the summer of 2017, the AHU operated as a SZVAV AHU for data acquisition without the incorporation of dehumidification control measures.

The AHU system operates in an occupied mode throughout the week, from Monday to Sunday, spanning the hours of 6:00 am to 6:00 pm. Within this timeframe, specific control parameters are implemented to regulate the system, ensuring optimal environmental conditions within the building. Concerning fan operation, the supply fan consistently operates. Specific measures have been implemented to control the supply air temperature. In cooling mode, the heating coil valve is closed, and the cooling coil valve is adjusted to maintain a preset Supply Air Temperature (SAT) within the range of 55°F/12.8°C to 72.5°F/22.5°C based on zone demand. In heating mode, the cooling coil valve is closed, and the heating coil valve is adjusted to maintain the SAT within the range of 72.5°F/22.5°C to 86°F/30°C based on zone demand. To control the supply air fan speed, the reset occurs between a minimum (10%) and maximum speed (50% in cooling mode, 30% in heating mode) based on zone demand. The minimum speed aligns with ventilation needs when the outdoor air damper is fully open, and the maximum speed is set for the required design

(a)



(b)

Figure 3.7 (a) 3D view of FLEXLAB Test cell X3A and (b) Schematic diagram of single-zone AHU in FLEXLAB

airflow for each mode. When not in economizer mode, the OA damper is fixed at a minimum position (10% to 15%), subject to resetting based on supply fan speed. Simultaneously, the return air damper is fully open, and the exhaust air damper is fully closed. Economizer mode activates when the outdoor air temperature is 3.6°F (2°C) lower than the return air temperature. During this mode, the OA damper opens to 100%, the RA damper gradually closes to 0%, and the EA damper gradually opens to 100%. In order to control the space temperature, the designated heating and cooling setpoints for the zone are maintained at 71°F/21.7°C and 74°F/23.3°C, respectively, throughout the occupied period.

3.4.2.2 Dataset description

As introduced in the previous section, we utilized the experimental dataset generated for an Air Handling Unit (AHU) with variable air volume serving a single zone in FLEXLAB at Lawrence Berkeley National Laboratory. In that experiment, faults were imposed on the control mechanism of the three main components of the AHU system: the outdoor damper and the valves of the heating and cooling coils. In the "stuck" scenario, the control signal values are automatically overridden with a constant value corresponding to the given conditions of any asset. In the "leaking" scenario, the bypass valve of the heating or cooling coils is activated to be open in a proportional position equivalent to the amount of leakage. In those fault scenarios, outdoor dampers are stuck either when they are "fully open" or at the "minimum position"; the heating coil valve is stuck either when it is "fully open" or "partially fully open," and it is leaking with an intensity of "40% of max coil valve flow"; and the cooling coil valve is stuck when it is "fully open" and is leaking with an intensity of "50% of max coil valve flow". Furthermore, normal conditions were also examined. Each scenario was tested throughout the day, with data points recorded at 1-minute intervals. Therefore, to cover all conceivable experimented conditions, the outputs are depicted in Figure 3.8 using a hierarchical fault tree, organizing the model hierarchically.

During the experiments encompassing various operational and fault conditions in the primary assets of the air handling unit, external parameter settings, and internal reactions governing the system's behavior and responses were documented as design inputs to create this dataset. The inputs, presented in Table 3.11, include air temperatures measured in outdoor air, supplied air, mixed air, and return air; the control signal adjustments for (i) the damper position of relevant air types, (ii) the valve position of the heating and cooling coils separately, and (iii) the fan speed of the supply air. Additionally, the supply air temperature heating and cooling set points are included. As only the occupancy mode is under investigation, the supply air fan statuses are consistently maintained as "on." This parameter is not evaluated as it remains constant throughout the experiment. Therefore, a total of 13 distinct design inputs are considered for constructing the predictive model.

Table 3.11 Design inputs for case 2

Input	**Description**
Supply Air Temperature	Measured AHU supply air temperature
Supply Air Temperature Heating Set Point	AHU supply air temperature heating set point
Supply Air Temperature Cooling Set Point	AHU supply air temperature cooling set point
Outdoor Air Temperature	Measured AHU outdoor air temperature
Mixed Air Temperature	Measured AHU mixed air temperature
Return Air Temperature	Measured AHU return air temperature
Supply Air Fan Speed Control Signal	AHU supply air fan speed; ranges from 0 to 1; 0 - fan speed is 0%, 1 - fan speed is 100%
Exhaust Air Damper Control Signal	Control signal for AHU exhaust air damper ranges from 0 to 1; 0 – damper should be fully closed, 1 – damper should be fully open.
Outdoor Air Damper Control Signal	The control signal for the AHU outdoor air damper ranges from 0 to 1; the 0 damper should be fully closed, and the 1 damper should be fully open.
Return Air Damper Control Signal	The control signal for the AHU return air damper ranges from 0 to 1; 0 – damper should be fully closed, 1 – damper should be fully open.
Cooling Coil Valve Control Signal	Control signal for AHU cooling coil valve ranges from 0 to 1; 0 – valve should be fully closed, 1 – valve should be fully open.
Heating Coil Valve Control Signal	Control signal for AHU heating coil valve ranges from 0 to 1; 0 – valve should be fully closed, 1 – valve should be fully open.

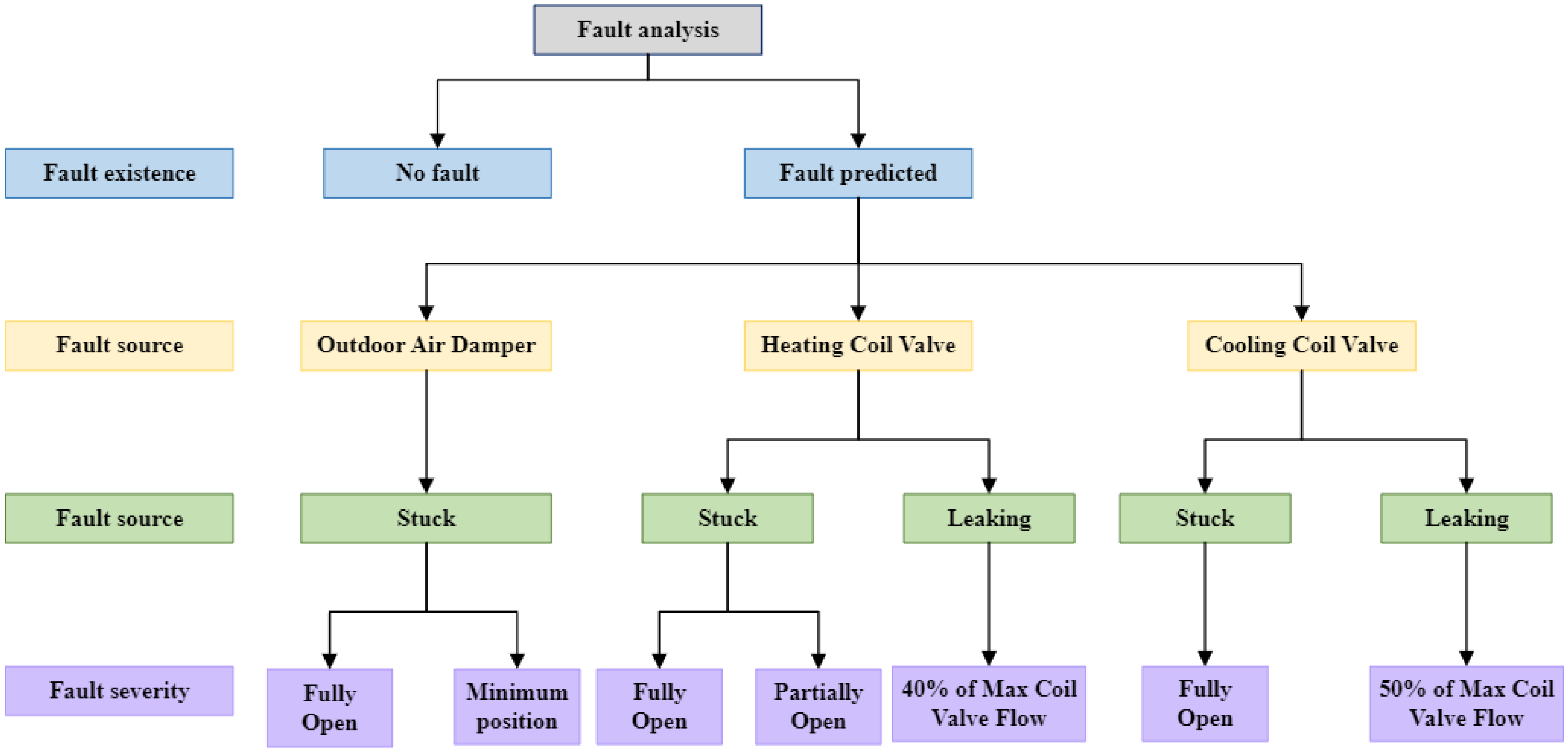


Figure 3.8. Fault hierarchy of case 2

3.4.2.3 Model description

Our hybrid intelligence approach is further validated using experimental datasets to demonstrate its continuous improvements. Following a two-step iterative process, similar to the procedure in the first case study, a systematic sampling approach is employed to group the data instances into five different sets. Each set comprises consecutive instances at fixed five-minute intervals, encompassing 576 instances. representing "no-fault" scenarios and 144 instances for each faulty class. Similar to the previous case, the design input data undergoes normalization, scaling the values within the interval of [-2, 2]. Through the two iterative step implementation, initially, a feedback-enhanced hierarchical model is trained with data from the first three groups, cross-validated reported in Appendix C, and tested with the fourth group. After that, the decision support model and human evaluations assess prediction accuracy. Feedback is incorporated by adding selected misclassified fault samples to update the model in the second step. The updated model is trained and tested with data from the fifth group, while the previous model is tested simultaneously for comparison, aiming to explore potential improvements.

A hierarchical model sequentially coordinates various local classifiers to analyze the operations of AHU. The goal of the model is to identify specific reasons if a fault is detected. The following local classifiers are proposed:

(i) In the first level of the hierarchy, a binary classifier is trained to predict whether a fault exists.

(ii) In the second level, a multi-class classifier is developed to classify the source asset of the faults,

(iii) As the outdoor damper has only stuck faults, no classifier is assigned to it. In contrast, two binary classifiers are trained separately to distinguish stuck and leaking faults for heating and cooling coil valves.

(iv) At the lowest level, two binary classifiers are proposed for stuck faults of the outdoor damper and heating coil valve to differentiate their positions.

Following the same procedure in the first case study, a confusion matrix utilizing the finalized predictions of the hierarchical model and ground truth conditions is generated for the training dataset. Additional local classifiers are then developed based on confusing class sets with a misclassification count exceeding the threshold value. The threshold value is set at 10. This feedback mechanism aims to improve the model's performance by addressing classes with a higher number of misclassifications.

The performance of both the initial and updated versions of the proposed model is evaluated and compared with the linear and nonlinear classifiers used in the previous case study.

3.4.2.4 Results

3.4.2.4.1 Performance analysis of hybrid intelligence approach

We examine the efficacy of our approach for the second case through a systematic, two-step iteration procedure. To begin with, our hierarchical model undergoes training employing a set of local classifiers, each specialized in a designated facet of the fault hierarchy. Within this model, the binary classifier excels in initially predicting the presence of a fault in the system, achieving a reasonable accuracy rate of 99.37%. At the next level, the model accurately detects all potential fault sources, such as the outdoor damper and the valve of heating and cooling coils. Advancing to the fault type hierarchical level, two different classifiers are trained to differentiate “Stuck” and “Leaking” faults of coil valves. The cooling coil dataset classifier precisely separates the decision boundaries between the two faults, while the other classifier faced challenges in distinguishing fault types for only two instances, yielding an accuracy rate of 99.85%. Further, two classifiers are employed to distinguish the "Stuck" position for the outdoor damper and heating coil valve. While the outdoor damper classifier accurately addressed all instances, the heating coil valve model struggled with differentiation in a considerable number of data instances. Consequently, our hierarchical model predicts the training data with eight distinct classes, achieving an overall accuracy rate of 99.10%. The model's predictions are evaluated against ground truth fault classes using a confusion matrix. Noteworthy challenges emerged in distinguishing between two specific class pairs:

(i) "0:No-fault" vs "4-2: Stuck outdoor damper in a fully open position."
(ii) "4-3: Stuck heating coil valve in a fully open position " vs "4-4: Stuck heating coil valve in a fully partially 50% open position ".

These pairs exhibit more misclassifications, with 15 fault-free instances erroneously predicted as faulty in the first case and 14 actual fault instances missing. In the second case, the stuck position of the heating cooling valve is mispredicted in 11 instances against nine fully open valves and two partially open valves in the ground truth. Therefore, implementing a feedback mechanism is imperative to enhance the model's accuracy in distinguishing these specific classes accurately.

The feedback solution is independently applied to the main hierarchical model as the two pairs have no common faults. Starting with the first pair, where the model misclassifies an instance with normal operations as "4-1: Stuck outdoor damper at minimum position", the focus is directed towards distinguishing faults in this specific pair. To provide comprehensive information gain, a binary classifier is implemented to differentiate between (i) "0: No-fault" and "4-2: Stuck outdoor damper in a fully open position". After training the binary classifier, referencing the fault condition, linear regression is applied to calculate the weights of the additional classifiers and the main model to improve the overall model. Within the optimization of the contribution weights as $w_{4,02}$=[0.7742 0.2332], the feedback classifier plays more role in the determination of the predictions; hence, the overall model decreases the number of false alarms and missing faults to 7 and 7, respectively, accurately predicting the more than half of the misclassified instances.

In the second pair, where the model encounters challenges in separating only two classes at the lowest hierarchy level, the architecture of the Neural Network is modified to offer value-additive feedback. Therefore, a binary Back-propagated

Neural Network classifier is employed, featuring two hidden layers, each comprising ten neurons activated by the sigmoid function. This classifier is specifically designed to classify the severity level of a heating valve stuck. Within linear regression, the feedback classifier dominates the process with contribution weights denoted as $w_{4,34}$=[0.9258 0.0751]. This refinement results in a noteworthy reduction in the number of misclassified faults to 7. Through these improvements, reported in Table 3.12, the accuracy rate of the model on the training dataset increases to 99.49%.

Furthermore, we conduct a performance assessment of the hierarchical model with a test dataset. The results revealed that the hierarchical model predicts fault classes with an accuracy rate of 95.83%. As anticipated, similar to the challenges faced during training, the model struggles with differentiating between pairs of the same class. Moreover, in contrast to the training set and cross-validation of the training dataset, a new pattern is identified, where instances of faulty conditions with a stuck cooling coil valve at a fully open position are erroneously predicted as fault-free. Given that the feedback model addresses only the first case, it corrects the testing results pertaining to these instances. Consequently, the number of false alarm instances decreases to 8, compared to 21 in the hierarchical model. Similarly, missing stuck damper faults decrease to 6 instances from the initial 12 cases before the feedback. On the other hand, under the dominance of the feedback classifier, incorrectly misclassified instances decrease to 3 (3+0) from 9 (5+4). Hence, this feedback corrects 25 predictions, resulting in an updated prediction accuracy of 97.41%, marking a 1.58% increase in total accuracy, as presented in Table 3.13.

Following the same procedure as in the first case study, the decision support model identifies 14 instances out of 41 as false predictions, which are then stored in the database. The remaining predictions are deemed suspicious and reported to the maintenance office through 6 work order requests. These requests consist of three false alarms, two missing faults, and one incorrect fault. Consequently, by utilizing both the direct predictions of the decision support model and the feedback from the site, a total of 34 misclassified samples are thoroughly evaluated and incorporated into the train.

Table 3.12 Confusion matrix of the training dataset trained by the initial feedback-enhanced hierarchical model for case 2

Training Set				Model								
				No-Fault	Outdoor damper		Heating coil valve			Cooling coil valve	
					Stuck		Stuck		Leaking	Stuck	Leaking
					Minimum position	Fully open	Fully open	Partially open 50%	40% of max coil valve flow	Fully open	50% of max coil valve flow
Actual	No-Fault			1720	1	7	0	0	0	0	0
	Outdoor damper	Stuck	Minimum position	0	432	0	0	0	0	0	0
			Fully open	7	0	425	0	0	0	0	0
	Heating coil valve	Stuck	Fully open	0	0	0	426	5	1	0	0
			Partially open 50%	0	0	0	2	430	0	0	0
		Leaking	40% of max coil valve flow	0	0	0	1	0	431	0	0
	Cooling coil valve	Stuck	Fully open	0	0	0	0	0	0	432	0
		Leaking	50% of max coil valve flow	0	0	0	0	0	0	0	432

Table 3.13. Confusion matrix of the test dataset trained by the initial feedback-enhanced hierarchical model for case 2

Training Set				Model							
				No-Fault	Outdoor damper		Heating coil valve			Cooling coil valve	
					Stuck		Stuck		Leaking	Stuck	Leaking
					Minimum position	Fully open	Fully open	Partially open 50%	40% of max coil valve flow	Fully open	50% of max coil valve flow
Actual	No-Fault			563	1	8	0	0	0	2	2
	Outdoor damper	Stuck	Minimum position	1	143	0	0	0	0	0	0
			Fully open	6	0	138	0	0	0	0	0
	Heating coil valve	Stuck	Fully open	0	0	0	140	3	1	0	0
			Partially open 50%	0	0	0	0	142	2	0	0
		Leaking	40% of max coil valve flow	0	0	0	0	0	144	0	0
	Cooling coil valve	Stuck	Fully open	15	0	0	0	0	0	129	0
		Leaking	50% of max coil valve flow	0	0	0	0	0	0	0	144

After updating the training set, all local classifiers of the hierarchical model are retrained, employing their previous versions as initial points to expedite the learning process. Consequently, the hierarchical model is reorganized using the updated local classifiers. In this new learning process, the updated hierarchical model demonstrates an improved accuracy rate of 98.91%, compared to the original model's accuracy of 98.62%. A detailed analysis of the confusion matrix indicates that the challenges faced by the model in the previous iteration persist for the same pairs. In the first case, the model incorrectly classifies 12 instances as faults, while it misses the faults for 22 instances by identifying them as fault-free conditions. On the other hand, in the second case, the model predicts seven instances of the heating coil valve being stuck in a partially open position when they are fully open and, in eight instances, vice versa. On the other hand, the behavior of unexpected instances in the previous iteration is learned by the model. As a result, the same binary classifiers are retrained to offer feedback to the updated model. Utilizing linear regression, the optimized contribution rates for the cases are quantified as $w_{u,4,02}$=[0.8301 0.1815] and $w_{u,4,34}$=[1 0]. This adjustment yields a significant enhancement in the overall model predictions: (i) the first feedback rectifies more than half of the misclassified instances, with a minimal occurrence of 4 false alarms and 11 missing fault instances; (ii) the second feedback further reduces the instances incorrectly classified to 4. Hence, as depicted in Table 3.14, which portrays the conclusive confusion matrix of the training set, the aggregate effect leads to an elevated prediction accuracy of the model, reaching 99.54%.

The overall model's performance is evaluated with new test instances. The hierarchical model demonstrates a notable accuracy rate of 98.58%, accurately predicting nearly all instances except one instance and conflicting pair instances. Those instances are 14 false alarms and ten missing faults in the first pair, and three and five misclassified faults in the second one, respectively. The incorporation of constructive feedback proves impactful. In the first case, false alarms significantly decrease, and a considerable reduction is detected in missing faults, as detailed in Table 3.15. On the other hand, within the second feedback, only two instances are

Table 3.14. Confusion matrix of the training dataset trained by the updated feedback-enhanced hierarchical model for case 2

Training Set				Model								
				No-Fault	Outdoor damper		Heating coil valve			Cooling coil valve	
					Stuck		Stuck		Leaking	Stuck	Leaking
					Minimum position	Fully open	Fully open	Partially open 50%	40% of max coil valve flow	Fully open	50% of max coil valve flow
Actual	No-Fault			1736	0	4	0	0	0	0	0
	Outdoor damper	Stuck	Minimum position	0	432	0	0	0	0	0	0
			Fully open	11	0	430	0	0	0	0	0
	Heating coil valve	Stuck	Fully open	0	0	0	435	2	2	0	0
			Partially open 50%	0	0	0	2	434	1	0	0
		Leaking	40% of max coil valve flow	0	0	0	0	0	433	0	0
	Cooling coil valve	Stuck	Fully open	15	0	0	0	0	0	432	0
		Leaking	50% of max coil valve flow	0	0	0	0	0	0	0	432

Table 3.15 Confusion matrix of the test dataset trained by the updated feedback-enhanced hierarchical model for case 2

Training Set				Model							
				No-Fault	Outdoor damper		Heating coil valve			Cooling coil valve	
					Stuck		Stuck		Leaking	Stuck	Leaking
					Minimum position	Fully open	Fully open	Partially open 50%	40% of max coil valve flow	Fully open	50% of max coil valve flow
Actual	No-Fault			575	0	1	0	0	0	0	0
	Outdoor damper	Stuck	Minimum position	0	144	0	0	0	0	0	0
			Fully open	6	0	138	0	0	0	0	0
	Heating coil valve	Stuck	Fully open	0	0	0	141	2	1	0	0
			Partially open 50%	0	0	0	0	144	0	0	0
		Leaking	40% of max coil valve flow	0	0	0	0	0	144	0	0
	Cooling coil valve	Stuck	Fully open	0	0	0	0	0	0	144	0
		Leaking	50% of max coil valve flow	0	0	0	0	0	0	0	144

incorrectly classified. Consequently, the overall accuracy of the tested instances improves to 99.37% for the overall model.

The decision support model systematically monitors predictions, discerning patterns in successive incoming instances. Out of 10 misclassifications, the model identifies five instances, documenting them in the database. Conversely, for the remaining instances, the model discerns contradictory patterns and initiates two work order requests to the maintenance office for further evaluation. These requests encompass one incorrectly predicted fault and one missing fault. On the other hand, the feedback mechanism prevents the maintenance team from one missing fault.

The efficacy of this approach is evaluated by comparing the performance of the original hierarchical model with the finalized model using the latest test instances. Results indicate that the original model achieves a prediction accuracy of 97.10%. In comparison, our model enhances the accuracy quite significantly by correcting 78% of previously misclassified instances, resulting in an overall accuracy of 99.37%. This improvement comprises a 0.82% contribution from regular classifier updates, and the feedback mechanism significantly contributes with a 1.45% enhancement in total. As highlighted in the previous case study, these enhancements reduce workload by decreasing the need for human intervention in decision-making. Hence, this underscores the robustness of our model in addressing the hierarchical fault reasoning problem.

3.4.2.4.2 Comparative analyses of FEHNNs with AI methods

We assess the efficiency of FEHNs by conducting tests on both the original and revised training and test datasets. We then compare its performance with widely recognized classification linear and nonlinear multi-class classification algorithms and hierarchical models. The following inferences are deduced from the results of the comparative analysis for both datasets, which are presented comprehensively in Tables 3.16 and 3.17:

Table 3.16. Prediction accuracy of different classification algorithms using original dataset for case 2

	Training Accuracy (%)	**Testing Accuracy (%)**
Logistic Regression	75.04	75.13
Support Vector Machines	79.34	79.55
Feed-forward Neural Network	93.39	92.55
Decision Tree	98.88	95.45
Hierarchical Decision Tree	99.03	95.71
Back-propagated Neural Network	98.78	97.29
Hierarchical Neural Networks	99.10	95.87
Feedback Enhanced Hierarchical Neural Networks	**99.49**	**97.41**

Table 3.17 Prediction accuracy of different classification algorithms using updated training set for case 2

	Training Accuracy (%)	**Testing Accuracy (%)**
Logistic Regression	74.99	75.57
Support Vector Machines	79.34	79.55
Feed-forward Neural Network	94.65	93.94
Decision Tree	98.93	96.53
Hierarchical Decision Tree	99.12	96.53
Back-propagated Neural Network	98.31	97.16
Hierarchical Neural Networks	98.91	97.92
Feedback Enhanced Hierarchical Neural Networks	**99.54**	**99.37**

- FEHNNs outperform all the algorithms under comparison, with the feedback mechanism notably enhancing the algorithm's accuracy compared to its hierarchical model, particularly in updated datasets.
- Linear classifiers, such as Logistic Regression and Support Vector Machines, prove unsuitable for this case, as they both inaccurately classify nearly a quarter of all test instances. Nonlinear classifiers improve their performance considerably in the initial scenario; however, the emergence of new patterns not covered in the trained model substantially reduces their testing accuracy. Despite this, with updates to the training model, they exhibit improved robustness in handling new test sets, except BPNN.

3.5 Research findings and discussion

In this study, we have developed a hybrid intelligence approach to ensure continuous improvement in detecting and reasoning faults within a system. The primary goal is to maintain a robust and resilient infrastructure, minimize downtime, and provide the system's ability to function efficiently and securely over time. The approach consists of six modules: (i) design module to construct the sensor network as design inputs; (ii) data acquisition and preprocessing module to collect data through real-time monitoring or simulation, clean and sample the data for training and testing purposes; (iii) predictive modeling module to select or design the right AI algorithm and train it for fault detection and reasoning ; (iv) monitoring and decision support module to test the constructed algorithm with incoming data, monitor consecutive predictions, and report suspicious patterns as work order requests to the facility maintenance office; (v) human-centric evaluation module proposed for maintenance team assesses the requests, conducts on-site inspections, addresses detected problems, and provides feedback to enhance prediction accuracy; (vi) continuous enhancement to utilize feedback from site inspections and the decision support model to enhance data support for fault detection and reasoning continually. In the proposed approach, BIM is integrated to interact with (i) the sensor network for extracting the relevant sensor information for the corresponding asset and (ii) the decision support model to provide spatial and maintenance information and documentation for the work order request.

In the design of the predictive model, we applied hierarchical classification, strategically breaking down the extensive set of fault classes into more manageable smaller units. This approach is particularly useful in systems with numerous fault types and varying severity levels across assets. The algorithm's structure aligns with the fault hierarchy, orchestrating multiple local classifiers in hierarchical harmony. It systematically checks for the fault's presence, identifies the fault's source asset, determines its fault type, and finally assesses its severity level. This structured approach aids in both interpreting and resolving issues effectively. Moreover, to address errors propagated through the hierarchy, we have developed local classifiers based on a Back-propagated Neural Network with conditional stepwise learning, mitigating the risk of overfitting. Additionally, additional local classifiers are trained to provide constructive feedback to update the main hierarchical model. This involves combined contribution weights, determined through linear regression, of the feedback classifier(s) and the main model, enhancing the predictive accuracy and robustness of the overall model.

As indicated in the previous section, the interaction between humans and machines is crucial for improving the predictability of fault reasoning in the development and sustainability of the approach. Humans play a significant role in the development of network construction, data preprocessing, predictive model selection, decision-making in fault management, and constructive feedback from site inspection to machine predictions to augment the machine and improve prediction accuracy. On the other hand, the machine shortens the repetitive and time-consuming processes in the modules and provides invaluable decision support through the prediction of fault reasons to facilitate human decision-making.

The applied case studies underline that the hierarchical fitness and feedback mechanisms to the predictive model and regular updates, incorporating feedback from test instances of the previous step, are the main drivers to enhance the approach's capability. With the contribution of the feedback provided from different sides, the accuracy of the predictive model increases, human interventions due to suspicious prediction patterns decrease, and both false alarms and missing faults are reduced. These improvements lead to an enhancement in the reliability of the approach in practice.

Besides the predictive model design and regular updates of training data, various factors affect the performance of the proposed approach. To begin with, the network of the sensors determines the design inputs of the predictive model. Although these inputs were predefined in the two case studies, in a newly constructed system, optimizing the location and type of the sensor is the first step. While the irrelevant sensors are removed through feature selection, missing ones could reduce model accuracy. Moreover, the accuracy of the approach is directly affected by the quality of sensor readings, influenced by fluctuations, gradual changes due to aging or environmental conditions, calibration issues, and sensor malfunctions, all contributing to noisy data. As a result, it is essential to detect such instances, ensuring they are not included in the training dataset and are excluded from consideration during testing to maintain the integrity of the evaluation process.

In addition, data quality and representativeness are paramount in ensuring the effectiveness and reliability of the predictive model. High-quality data, free from errors and inaccuracies, is the foundation for training models to make accurate predictions and decisions. Moreover, the representativeness of the data is crucial for enabling the model to create generalizable solutions to incoming and unseen scenarios. If the training data is biased, incomplete, or unrepresentative of the real-world context, the model may struggle to handle diverse inputs and can lead to skewed or unfair outcomes. On the other hand, from the point of initial training, developing the predictive model is challenging due to the unavailability of sensor data. This problem can be mitigated by following the strategies:

(i) Simulating data that represents expected patterns and behaviors of faults. This synthetic data can serve as an initial training set for the model. It may not capture the complexity of real-world scenarios, but it provides a starting point for the model to learn.

(ii) Utilizing pre-trained models on a similar system and fine-tuning them with synthetic or limited real-world data. This transfer learning leverages knowledge gained from other sources, potentially accelerating the learning process if it is sufficiently representative.

(iii) Collaborating with domain experts to gain insights into the expected behaviors of fault conditions, using their expertise to detect and label fault types manually during data collection. However, this effort is not expected to cover all faulty conditions sufficiently.

(iv) Testing faults in controlled environments with minimal risks. This allows the facility maintenance team to observe its behavior, identify potential issues, and iteratively improve the model before deploying it in more complex or critical scenarios.

In practice, a preliminary predictive model is established by employing one or a combination of strategies to provide a solid starting point. As real-time data accumulates with accurate labeling, the initial training dataset is regularly updated by replacing or supplementing it with authentic, representative data. This ongoing process ensures that the models are continuously trained with the most relevant information, leading to increased accuracy and adaptability over time.

Furthermore, data preprocessing is a vital step in achieving an accurate and reliable predictive model. It consists of multiple techniques to prepare the raw data for optimal training:

(i) symmetric normalization of design inputs, such as [-2, 2], to scale the data for allowing all variables to contribute proportionally to the learning process,

(ii) feature selection or elimination to enhance model efficiency and accuracy by retaining only the most relevant features,

(iii) removing noise, handling missing values, and addressing outliers.

Therefore, these preprocessing steps not only mitigate potential biases but also contribute to improving the model's interpretability, generalization, and robustness, resulting in more precise predictions and better overall model performance.

As stated in the model initialization, transfer learning utilizes the accumulated knowledge from the previous cases to expedite the learning process for updating the model. In this study, we implement the classifiers trained in the last iteration as the starting point to iteratively update the model through regular updates. This approach leverages the information learned in the previous iterations, enhancing the efficiency and adaptability of the model to evolving data patterns over time.

According to the findings presented in the case study, FEHNN's performance surpasses that of widely recognized linear and non-linear classifiers. Consequently, ensemble models emerge as viable alternatives for enhancing our model's competitiveness since they combine the predictions of multiple base models to compensate for the weaknesses of individual models, improving overall performance, generalization, and robustness. However, understanding the contribution of each base model to the overall prediction might be less straightforward compared to individual models. Therefore, three different ensemble model types are proposed in the literature. These models and their comparison with our solution approach are explained as follows:

(i) bagging model to train multiple instances of the same base model on different subsets of the training data and combine their predictions through averaging or voting to improve overall model robustness. In MATLAB, Decision Trees are employed by default to train these models using the randomly selected subsets of the updated training dataset to construct Random Forests in the second case study, using *fitcensemble(input,output,"Bag")*. The analysis reveals that the bagging model RF markedly enhances the training accuracy to 99.79%; however, its test accuracy is notably lower at 98.61% compared to our model. Due to the use of numerous base models in composing the overall model and the inherent flexibility of Decision Trees, this approach leads to both overfitting and challenges for the maintenance team in interpreting prediction results.

(ii) boosting models to train the multiple weak learners sequentially, where each learner corrects the errors of its predecessor. The final prediction is a weighted combination of the individual models. This model is tested with AdaBoosting, utilizing MATLAB classifier *fitcensemble (input, output, "AdaBoostM2")*, but it decreases the model's accuracy by under 90%.

(iii) stacking models to combine the predictions of multiple diverse base models trained using the same datasets, using a meta-learner. The meta-learner is trained on the predictions of the base models. In this study, Decision Trees is implemented as a meta-learner using the predictions of BPNN and HNN as inputs to the model. The results demonstrate that the meta-model enhances the performance of both individual models in terms of training and test accuracy, achieving rates of 99.64% and 98.36%, respectively. However, compared to our proposed feedback mechanism, the stacking model struggles to effectively address overfitting, leading to limited improvement in testing accuracy despite high training success. Additionally, the nonlinearity introduced by both individual models and the meta-learner results in a complex model, making it challenging to understand the system behavior. Alternatively, we explored combining the predictions of BPNN and HNN through a process that optimizes their weight contributions using linear regression, mirroring our feedback mechanism. Notably, the distinction lies in our approach: our feedback mechanism focuses on resolving the main conflicting points, whereas the stacking model considers the complete set of predictions. In contrast to the nonlinear approach, this method is more interpretable. Nevertheless, it comes with a drawback of less accurate predictions, as indicated by accuracy rates of 99.00% for the training set and 98.17% for the test set.

The comparison between our model and ensemble solutions reveals that the suggested feedback mechanism substantially improves the performance of the hierarchical model, providing robust and competitive solutions. Moreover, this mechanism can be extended to multi-class classifiers, aiding in clarifying distinctions between frequently misclassified classes.

In this research, when tackling the problem of fault reasoning classification, we treat all fault classes as equally important. Nevertheless, in practice, the impact of faults on the system and facility varies. Faults in critical assets can lead to system breakdowns, while others may only slightly decrease system efficiency. As a result, the maintenance team aims to address prominent issues since their misclassification costs significant consequences, while less critical faults may be less pressing. Therefore, weighting schemes can be utilized to assign different weights to different classes in the loss function of the model (in Eq 3.1.) to address imbalances and improve the model's accuracy, especially when certain classes are underrepresented or have more significant consequences.

Following the model training, the decision support model examines prediction patterns in real-world operations to detect faulty conditions and identify suspicious patterns. While it is not properly designed, it may miss false alarms, or the existing fault may not be detected when it is omitted. Moreover, it may bring a significant workload to the facility maintenance office by frequently reporting numerous conditions as suspicious patterns. During this process, the maintenance team may identify patterns that challenge the predictive model's differentiation abilities. Consequently, this information is provided as feedback to the model for incorporation into regular updates. Despite the updates, certain fault patterns known to the maintenance team may remain undetected by the predictive model. This diminishes the reliability of both the model and how the maintenance team perceives it. Therefore, it is quite critical to construct a bridge between the predictive model (machine) and maintenance office (human) for managing the prediction analysis efficiently. Initially, the facility maintenance office formulates patterns using a set of rules for the model to detect and comprehend. These rules are revised when repetitive patterns are flagged as suspicious or when the predictive model gains an understanding of conditions through regular updates. As a result, it is designed to determine patterns, report the detected faults as work order requests, and minimize the recognition of suspicious patterns so that the maintenance office can be left with minimum effort. In other words, humans play a supervisory role in the control and confirmation of the work order request and intervene only in edge conditions for decision-making, and revision is needed according to the feedback provided by the site inspections. Moreover, the decision support model collects representative unusual data for fault-free and faulty conditions to facilitate the generalization ability of the updated models. Utilizing the collected dataset and feedback from on-site inspections for misclassified cases, the model undergoes regular updates by resampling the training data. This iterative process ensures continuous refinement, allowing the model to adapt to emerging patterns, enhance its accuracy, and improve reliability based on real-world observations and practical feedback. Once this approach is established and operational over time with human-machine collaboration, initial human intuition on the construction of modules is replaced by machine automation to facilitate and standardize the processes. On the other hand, humans take the place of final decision-making, site inspections, and operational controls of the overall approach.

Within this approach, faults are categorized into classes and predicted through the utilization of a feedback-enhanced hierarchical model in system operation. This model is trained using readings from sensors installed on the system assets. Nevertheless, in certain scenarios, fault detection and isolation become more straightforward by implementing threshold rules based on the readings from the relevant sensors. Hence, by filtering out these readily detectable faults, the model directs its attention more towards faults exhibiting nonlinear behavior, narrowing down to a reduced set of fault classes. In this scenario, faults are initially categorized into two groups under the guidance of domain experts based on their ease of detectability. Threshold rules are established for each conspicuous fault, while the remaining faults are organized according to the fault hierarchy. Our model is then trained on these fault classes using their respective training datasets. During the assessment of incoming instances in operational testing, threshold rules are initially employed on the instance. Upon detecting a fault, the decision support model promptly generates a work order request, reporting it with certainty to the facility maintenance office. Conversely, if no fault is identified, the predictive model is activated to estimate the system's condition. For this case, the decision support model applies the same procedure in the original approach for predicted instances.

In this study, our classification model is developed to detect the reason for a fault if it exists. It is limited to the detection of a single fault at a time; however, in practical operations, multiple faults may occur simultaneously. In this scenario, the model checks whether a fault exists and then predicts the source and type of each fault. When multiple faults occur simultaneously, they may appear either in the same asset with different fault types, in different assets, or in a combination of both. In simpler terms, although the local classifiers for fault detection and severity level classification remain unchanged, the ones for fault source and type classification are adjusted to accommodate the selection of at least one class for the same instances rather than just one. In the light of this information, the model can be modified in two ways:

(i) in the first approach, the algorithmic structure of the original hierarchical model is retained. However, modifications are made to the local classifiers responsible for determining fault source or fault type. In those cases, the softmax activation function in the output layer of multi-class classifiers is replaced by the sigmoid function. Additionally, binary classifiers are substituted with two-class (multi-class) classifiers, utilizing the sigmoid activation function in their output layers. Consequently, all classes are activated for potential selection. The training of these classifiers involves using the loss function specific to a multi-class problem, as outlined in Eq. 3.1. On the other hand, in the feedback mechanism, the procedure applied in the original mode is followed to update the model. The only variation occurs during the training of the feedback classifier, where instances involving at least two of the compared fault classes are excluded from the training set

(ii) in the second approach, the hierarchical level between fault source and fault type is deconstructed, and a horizontal level based on fault type, which incorporates source information, is established. Hence, the model first checks the fault presence then predicts the existing fault types of the sources if a fault exists, and finally determines the severity levels of each. As in the original hierarchical model, the same classifiers are employed for fault existence and severity level predictions. Conversely, in the second level of the modified fault hierarchy in this model, a binary classifier is designated for each option. These classifiers are trained with faulty datasets, employing a one-vs-all approach to compare the selected faults with the remaining ones. This is done to determine whether a fault is detected for the respective fault type in the asset.

Additional binary classifiers are introduced for conflicting classes in the feedback mechanism to enhance the algorithm's performance. These classifiers are specifically built upon the more critical asset(s). In the implementation of successive feedback for the same fault type, fault-free feedback is initially utilized, followed by the introduction of faulty instances based on their criticality.

In contrast to the original model, the algorithmic structure of the multi-fault models accommodates some conflicting issues. For example, while the initial classifier detects the faults for an instance, in the hierarchy levels of fault source and fault type, none of them predicts its occurrence. Hence, the detected fault might not be sustained in the next levels, leading to either the rectification of false alarms or the missing of the faults at lower hierarchy levels. Therefore, our model, including an enhanced feedback mechanism, also tackles this issue to improve the model.

3.6 Conclusion

In this research, we proposed a hybrid intelligence approach that relies on human-machine interaction to improve fault detection and reasoning within a system continuously across its life cycle, addressing RQ 2. Within the approach, humans are pivotal in tasks such as network construction, data preprocessing, model selection, decision-making in fault management, and offering constructive feedback from site inspections to improve machine predictions. Simultaneously, the machine streamlines repetitive processes and provides crucial decision support by predicting fault reasons, ultimately facilitating human decision-making. Therefore, through this collaborative and synergistic approach, our solution attains sustainability with continuous improvement. Moreover, BIM is integrated to interact with the sensor network and decision support model, providing spatial and maintenance information for efficient work order requests.

In constructing the predictive model, a hierarchical classification approach was employed to break down an extensive set of fault classes into more manageable units, particularly beneficial for systems with diverse fault types and varying severity levels. The model aligns with this fault hierarchy, orchestrating multiple local classifiers for systematic fault detection, source asset identification, fault type determination, and severity level assessment. To address potential information losses, especially in the hierarchy, local classifiers based on a Back-propagated Neural Network with conditional stepwise learning were developed, mitigating overfitting risks. Moreover, Additional local classifiers contribute feedback through combined weights determined via linear regression, enhancing the overall model's predictive accuracy and robustness.

In essence, this research contributes to advancing fault detection and reasoning approach by combining human and machine intelligence, utilizing hierarchical models with feedback mechanisms, and discusses challenges in data quality, representation, imbalanced distribution, hybrid prediction models, multi-fault scenarios, and evaluation of suspicious prediction patterns. The proposed approach offers a practical and effective solution for maintaining resilient and efficient infrastructures over time, with continuous improvement driven by collaborative efforts between humans and machines.

The performance of the approach was validated with two existing case studies with a relatively limited number of fault classes throughout one-stage improvement. In comparison to classifiers previously validated for their efficiency in those studies, the approach exhibits superior performance. In future studies, the robustness of the approach will be investigated with more complex cases involving a greater variety of fault types and severity levels across a more extensive set of assets.

CHAPTER 4

A FRAMEWORK FOR MODEL-BASED WORK ORDER MANAGEMENT AND FAULT NETWORK ANALYSIS APPROACH FOR FAULT REASONING

4.1 Introduction

Efficient work order management is essential for the well-organized and successful implementation of maintenance tasks in a facility. It provides the systematic creation, assignment, tracking, and completion of work orders related to equipment, facilities, or assets that require maintenance. The process begins with identifying maintenance needs, which are then translated into work orders specifying the tasks, resources, and timelines. Work order maintenance management helps prioritize and schedule maintenance tasks efficiently, ensuring that critical equipment or systems receive prompt attention. It also facilitates communication between maintenance teams, supervisors, and other relevant stakeholders, fostering collaboration and accountability. By centralizing and streamlining maintenance processes through work orders, the facility can optimize resource utilization, reduce downtime, extend the lifespan of assets, and enhance overall operational efficiency.

Integrating Building Information Modeling with work order management is a powerful strategy that brings digital intelligence to the maintenance processes. By linking BIM data to work orders and utilizing the model, maintenance teams can (i) access detailed information about the structure, components, and systems involved, (ii) visually analyze the maintenance requirements to enable more informed decision-making, (iii) collaborate and communicate more efficiently and streamline information sharing and decision-making processes to execute maintenance tasks more efficiently; (iv) leverage BIM's capabilities to anticipate potential issues and schedule preventive maintenance proactively.

This integration allows for better visualization of maintenance requirements, enabling more informed decision-making. For example, maintenance personnel can quickly identify the location of a specific asset within a building or facility, understand its history, and access relevant documentation. Additionally, it enables real-time collaboration and communication between stakeholders to foster better coordination and information sharing and streamline decision-making processes to execute maintenance tasks more efficiently. Moreover, BIM's predictive capabilities can be leveraged to proactively anticipate potential issues and schedule preventive maintenance. By analyzing the model, the maintenance team can identify assets requiring attention based on their condition, usage patterns, or historical performance data.

Utilizing a work order management system to facilitate fault reasoning is paramount for the seamless integration of maintenance processes. By serving as a centralized repository for fault reporting and resolution, it ensures that reported issues are systematically documented, creating a structured database for fault reasoning. Moreover, real-time integration with monitoring systems enables immediate fault detection, triggering automated work orders and facilitating a proactive approach to maintenance. The system's workflow automation expedites fault resolution and ensures a systematic reasoning process. The historical records provide insights into fault patterns, aiding maintenance teams in root cause analysis and long-term decision-making. Although it facilitates the maintenance workflow and enhances the decision-making processes, the efficient utilization of the historical records in the workflow, especially finding the root-cause patterns, depends on the quality and understandability of the provided information. Therefore, in this research, we investigate the isolation and reasoning of the faults using work orders integrated with the BIM environment. First, the case studies from the practice and literature studies on BIM-based reasoning of the faults are examined, then, model-based work order management framework and fault network analysis approach to isolate the fault and pinpoint the reason, if available, are presented and validated with case studies.

4.2 Motivating case studies on work order management in practice

We have conducted two informative motivation case studies to better understand the current statement of work order management in practice, specifically focusing on information accessibility and retrieval. In the first case, we examined work order forms and BIM-integrated and standalone CMMS tools to analyze the attributes of work orders and gather valuable insights. On the other hand, the focus of the second case study was a more in-depth concentration on the work order records stored in an airport work order database. This investigation specifically delved into the fault descriptions and explanations of performed work, intending to retrieve beneficial information to establish a linkage between them for facilitating fault reasoning.

In the first case study, we divided the work order documentation into two parts: one for work order requests and another for tracking and reporting. A comprehensive analysis was conducted on 46 request forms, revealing that common attributes such as the date of request, location, requester name, and problem description were consistently present. Notably, approximately three-fourths of the forms encompassed contact information. On the other hand, 12 forms showed how urgently the customer needed maintenance work. Regarding the identification of the source asset causing the problem, only six forms directly addressed this issue. Additionally, merely two forms sought information on the maintenance history of the same problem. Finally, a single form provided a comprehensive selection of options in a checkbox format for problem types, including categories such as cold, hot, leakage, clogged, cracked, and smells. Furthermore, out of the 46 forms analyzed, details regarding the tracking and reporting of the work order were present in only 17 forms, whereas the remaining ones were reported in a separate document. While all forms consistently included details about the assigned person for the maintenance task and its completion date, the work order action description was absent in one-third of these request forms. Conversely, it consistently appeared in all the separated forms; however, only three forms asked for the root cause of the problem in the attached action plan. Moreover, the approvals of the completed maintenance tasks were available in 21 forms, while the detailed scheduling and tracking tables were presented in only four forms. The statistics of these forms are reported in Table 4.1.

Table 4.1 Statistical information of analyzed work order and request forms

Information details	The number of forms corresponding information is available
Analyzed request form	46/46
Information about the requester's name, request date, location, and problem description	46/46
Contact information of the requester	34/46
The urgency of the requested problem	12/46
Source of the requested problem	6/46
Questioning the maintenance history of the requested problem	2/46
Categorization of the requested problem	1/46
The form includes both request and work order details Information about the assigned person and completion date of the work order Work order description	17/46 17/17 11/17
Work order and request forms are separate Information about the assigned person and completion date of the work order and work order action description	29/46 29/29
The root cause of the problem reported in the action plan	3/46
Approvals of the completed maintenance tasks	21/46
Detailed scheduling and tracking table	4/46

Similarly, we applied the same procedure to analyze the attributes of work orders and their requests in CMMS tools. The investigation focused on the work order generation process across 12 different CMMS tools, including Maximo, Ecodomus, Limbe, QRmaint, Mobility Work, MaintainX, UpKeep, Fiix, MainWinWin, AssetPanda, FM Systems, and BIM Genie, reported in Appendix D. While most tools are primarily designed for essential work order management, a few offer more advanced solutions. To begin with the work order request, essential CMMS tools replicate request forms, providing content such as the requester's details, request description, and location information. In contrast to the forms, these requests are enhanced with additional files and photos from the site. Conversely, advanced CMMS tools enrich request content with information about the asset requiring maintenance, the requester's priority, maintenance type, problem type, and the requester's expected schedule. Compared to basic forms, both location and asset can be selected from predefined lists within the tool. This functionality is particularly noteworthy in BIM integration, where it can be presented within a visualized environment. In work order tracking and reporting, the basic tools report the work order description, its priority, asset, location information, the assigned person to handle the maintenance task, and schedule information. The advanced tools focus on more detailed aspects of the work order. Specifically, they (i) separate the description of the problem into two parts, fault description and works performed; (ii) categorize the work orders based on maintenance type, failure location or system (HVAC, pipe, runway), and problem types (temperature, lightning); track the status of the work order, including its planned and actual schedule; and report the responsible personnel for the maintenance task and its progress tracking for approvals. Moreover, they consolidate relevant work orders under a parent work order umbrella for consistent evaluation. By linking maintenance documentation of the assets with the work order, safety plans, specifications, and contract issues can also be assessed in conjunction with the work order. Furthermore, among the 12 tools, only three directly report the textual root-cause attribute for the work order.

The investigation of the attributes of the work orders for request, tracking, and reporting shows that the forms limit the attributes to gain information from the maintenance request and its post-solution reporting. It is more related to reactive solutions, where the effectiveness of time utilization and access to on-hand information heavily relies on the maintenance worker's experience. However, lessons learned from that work order from each work order are confined to the knowledge of the specific worker involved. Hence, personnel turnover may lead to the loss of valuable information accumulated from previous cases. When a new worker faces similar challenges, he must invest additional time to attain the same level of experience as his predecessor in an identical case. Therefore, either recording the filled forms in a CMMS tool or directly using the CMMS tool to create work orders establishes a database. This database serves as a repository for retaining historical records, ensuring the preservation of information gained at least. By adopting this practice, the maintenance team protects valuable insights, even during personnel turnover, facilitating efficient knowledge retention across various maintenance cases. Within the maturity of CMMS, it enables more comprehensive and structured work order recording and organizes the maintenance documentation more systematically. This allows the utilization of historical records for work order analytics, examining the distribution of work orders based on location and assets. Moreover, the analyses also include performance metrics such as completion times, resource utilization, and costs, and businesses can make informed decisions about workforce management, equipment maintenance, and budgeting to improve the maintenance processes. The integration of Building Information Modeling (BIM) further contributes by visualizing this distribution through color coding, enhancing understanding of the current status within the facility. Additionally, BIM strengthens maintenance task documentation and information management. It also simulates and analyzes scenarios, aiding in identifying potential risks and challenges to address issues before escalation, reducing the likelihood of emergencies and associated work orders. BIM's role extends to managing spatial relationships within the facility, ensuring that work orders align with the building's physical layout for efficient work order scheduling. On the other hand, despite the improvements in the process and structured information of asset and location information, the most critical information about the details of the maintenance task is provided from the description of the work orders (what the problem is) and works performed to handle the issue (how it is resolved). Therefore, the quality of the textual information offered in these two attributes of the work order might enhance the efficiency of the next maintenance tasks.

In the second case study, we examined the descriptions of the historical work order records collected throughout the airport maintenance. The airport maintenance office had delegated maintenance tasks for distinct sites to different firms under clearly defined contracts. All corresponding work records were systematically collected in a centralized maintenance database. In the airport, a significant portion of tasks underwent regular inspections, while corrective maintenance was implemented for the remainder. The work orders were classified according to their asset type or location in the given database. In this case, we analyzed the work orders in the corrective maintenance database and those for preventative maintenance where tolerable faults were identified during inspections or significant anomalies were detected suddenly for the regularly assessed assets. Out of 2,884 work orders documented in the preventative maintenance database, 266 records revealed anomalies, while 805 corrective work orders were examined.

To begin with corrective work orders, first, we examined work order descriptions on faulty conditions. The findings revealed a lack of clarity in problem descriptions in numerous cases. Specifically, in 156 work orders, the only explanation provided was that the asset was "not working." Similarly, for lights, "not working" was replaced by "off" in 279 instances. Additionally, there were instances where the same issue was articulated using different phrases, such as "lights were not working" or "lights were off." Furthermore, in 223 orders, the issue was conveyed through the general term "need," as seen in statements like "needs a carpenter" or "the asset needs to be fixed." However, these explanations lack clarity. In the first case, the reason for needing a carpenter is not clearly described. In the second case, the entry merely acknowledges the presence of an anomaly in the specified asset without providing detailed information. As a result, the explanations fall short of providing the clarity necessary to extract valuable insights to enhance the maintenance process. On the other hand, an examination of the descriptions detailing the actions taken to resolve issues in the specified work orders yielded the following results: (i) in 288 instances, no information was available as the field was left blank; (ii) 40 instances revealed false alarms, indicating no problems were found during site inspections; (iii) in 58 instances, explanations were present, but they lacked substantive information such as "work is completed" or "the issue is fixed." Conversely, in 320 instances, the faulty asset was directly addressed using terms like "fix, replace, install, and change"; however, this information was also available since the source asset was an attribute of the work order. Thus, out of 805 work orders, the descriptions of the performed works provided additional information in only 67 instances, encompassing detailed explanations of the fault source along with its corresponding 45 faulty subparts and 22 root causes of the problem.

We also analyzed 266 work orders addressing the faults in the preventative maintenance database regarding fault description and works performed. Beginning with fault descriptions, in 177 instances, the explanations of the fault conditions were overly broad, providing insufficient clarity. This included phrases such as "not working" in 167 instances and "need …." in 13 instances; however, compared to the corrective work orders, the faults were more clearly explained in these instances, such as “leaking in an asset” or “ the asset is not cooling.” On the flip side, an investigation of the descriptions explaining the work performed to handle the problems in the specified work orders revealed the following outcomes : (i) in 39 instances, the field was left blank, resulting in a lack of available information; (ii) in 58 instances, the explanations were inferior, lacking substantial information. (iii) in 188 instances, the problematic asset was explicitly mentioned using terms such as "fix, replace, install, and change." However, it's worth noting that this information was redundantly available since the source asset was already an attribute of the work order. Therefore, only 73 instances contributed valuable information, presenting detailed explanations of the fault source, encompassing 41 corresponding faulty subparts and 32 root causes of the problem.

The following inferences were drawn from the examination of the description entries in the work order records, encompassing both corrective and preventative maintenance tasks:

(i) While the descriptions in preventative maintenance tasks are notably clearer compared to corrective ones, only one-third of their fault descriptions provide valuable input for information management in maintenance. Conversely, 27% of these work orders yield additional information to the process, with only 12% of them addressing information about the root cause of the problem. It was lower than 10% in all three cases for the corrective maintenance.

(ii) The content and format of the description entries exhibit variations from worker to worker and subcontractor to subcontractor.

(iii) Integrating the work orders with the BIM model solely offered a visual representation of faults and their statistics; however, its spatial and relational repository did not interact with the work orders and their interpretation.

The findings from these case studies indicate the necessity for (i) customizing the work order request and reporting section to enhance clarity in information retrieval and root-cause analysis; (ii) standardizing the format and content of description entries for fault description and work performed; and (iii) leveraging Building Information Modeling (BIM) not only for visualizing work orders but also for establishing relationships between problems and corresponding resolutions using its spatial and relational repository, expediting the maintenance process accurately. The research outlined in this chapter addresses these requirements by developing a model-based work order framework and proposing a fault network analysis to capture information gained in tasks and enhance the reasoning mechanisms for faults, thereby improving their utilization in future tasks.

4.3 Background research

The research efforts on refining the reasoning mechanism within Building Information Modeling for facility maintenance have been a focal point in previous studies. Akcamete et al. (2011) emphasized the significance of maintaining an up-to-date model for an accurate reasoning mechanism in facility maintenance. To achieve this, customized templates were created within the model to record facility changes. Utilizing these templates facilitated the establishment of a change history, aiding in identifying patterns in repetitive maintenance tasks.

In a related study, Lucas et al. (2013) proposed a research study aimed at developing a healthcare facility maintenance information management system. The study delved into how facility failures impact patient health to develop an efficient maintenance strategy. Through case scenarios, potential failure modes were generated. The fault tree analysis and failure mode and effects analysis were employed to determine the causes by retrieving data of the assets in the model and the effects of the failure in the facility, respectively. Considering the source of the failure, an analysis was conducted to understand its impact on patients and to identify the necessary actions required to minimize or mitigate the failure.

Motawa and Almarshad (2013) introduced a knowledge-based Case-Based Reasoning (CBR) model linking root-cause relations between problems and model elements through previous case studies that included problem, solution, and associated model elements affected by the case. The model employed a manual structured filtering mechanism to define the issues and filter similar studies based on their similarity to the defined problem. Model elements were utilized solely to connect cases with relevant elements.

Motamedi et al. (2014) proposed a knowledge-assisted BIM-based visual analytics for root-cause detection in facility faults and condition-based maintenance. Faults or conditional statements were identified using CMMS work orders, conditional maintenance, and inspection reports. The model visualized results in the BIM environment through color codes, associating problems with the model elements and utilizing fault tree analysis to determine potential sources of issues.

Yang and Ergan (2016a) developed a framework for identifying and retrieving information to troubleshoot HVAC-related corrective maintenance problems. Considering the HVAC-related problems in practice, 40 work orders were generated based on characteristics such as reported problem type, spatial scope, HVAC system type, HVAC control system type, and time pattern of problems. For each work order, applicable causes were listed. Using the model, the characteristics of the HVAC system and reported problems were matched with work orders to filter possible applicable causes of HVAC-related failures.

Alavi and Forcada (2022) developed a hierarchical rule-based framework to isolate the location of the faults and determine the root cause of HVAC problems reported by occupants. According to the investigated rules, room and asset information is extracted to compare the existing thermal load of the room with the asset’s indoor capacity. When an HVAC design problem is detected, the rules question the building envelope performance and the indoor capacity of the HVAC system successively; otherwise, information about the pressure and temperature from the Building Management System is checked to decide whether the outdoor or indoor unit of the HVAC system causes the discomfort problem. To implement this procedure, the framework was integrated into the visual programming tool Dynamo, which directly interacts with the BIM model. Consequently, the framework extracts maintenance requests from the CMMS tool and temperature and pressure values from the Building Management System to apply the rules. Dynamo identifies the HVAC system in the reported room, extracts the indoor capacity value from the model asset's features, and executes the procedure. Finally, it visualizes the root cause of the reported work order using color coding. This study demonstrates the use of BIM to gather non-geometrical information about an asset from the repository and visualize the detected problem's root cause.

In the previously mentioned research studies, the relationships between the problem source and its influencing model elements were established based on pre-defined customized templates and case studies. Each study constructed a database for predetermined situations. Therefore, for the current problem, algorithms were employed to search and filter templates in the database, ranking or filtering them based on relevance. Following this, a developed root-cause mechanism associated with the selected template was utilized to identify the source of the problem and its correlated model elements. These model elements were only used to be associated with the selected template. Furthermore, the focus on intelligent data was limited to filtering applicable root-cause options by matching the properties of model elements with those constructed in the historical or created template database. This approach leaves a gap in utilizing intelligent model data to automatically establish relationships in the course of root-cause analysis, incorporating spatial and element/system-specific properties.

4.4 A model-based work order management framework

In this study, we designed a model-driven framework for managing work orders, aimed at structuring the content and format of work orders to establish connections among observable symptoms, the source asset of the fault, spatial information, and the impacted assets within the BIM environment. By leveraging automatically detectable relations, this framework utilizes information gained from historical work order records to streamline the isolation of faults and their root-cause relations. Therefore, using the BIM model, the framework reduces the time needed to identify and address the root of the problem and resolve the issue. Additionally, the maintenance team can promptly access the necessary maintenance information because they are familiar with the organized structure facilitated by the model. Hence, the systematic organization of maintenance information and documentation enables efficient retrieval whenever needed.

In this framework, work orders are generated with a focus on space-centered information. Hence, when submitting a work order request, the occupant first specifies the space where the fault is detected. The remaining details are then structured and organized based on the identified space. Therefore, the relationship types established between the assets in the model are defined based on the space-centered perspective as follows:

(i) Spatial relations are defined to pinpoint assets within the relevant space and determine their positions, while also outlining the physical contacts between them. Furthermore, spatial relations encompass the vertical projection of assets from one floor to the floor below, particularly when analyzing faults related to liquids. This relational framework also considers the interconnection of spaces in terms of proximity, adjacency, or any spatial configuration.

(ii) Systemic relations encompass interconnected elements that operate cohesively within a specified system. These relations denote functional dependencies, indicating that the performance or behavior of one component can impact or rely on another component within the system. To be more specific, the occurrence of an anomaly in a system's asset can lead to issues within the associated space where another asset of the system is located.

(iii) Feature-based relations are employed to classify the substitutable assets or systems on shared features and functionality. Hence, problems identified in one asset can be leveraged to detect similar issues in another asset.

The focus of the framework is limited to the content and format of the work order and their interaction with the BIM model to make work orders more intelligent and leverage acquired information for expediting fault isolation and reasoning. Therefore, certain typical aspects of a work order management system, such as scheduling and prioritizing work orders, optimizing resource utilization, integration with other business systems, and related considerations, are not directly addressed within the current framework. The framework prioritizes efficiency and intelligence in fault handling through its specific emphasis on work order content and BIM model interaction.

The work order management framework comprises four modules: work request, evaluation and management, site feedback, and data analytics. The framework utilizes Dynamo visual programming to facilitate the interaction between work order content with the BIM model. The Dynamo interface serves as the interactive input platform for the requester and site maintenance team, allowing seamless integration of their content into the system. Hence, the work request module processes the occupant's complaints by directing them to comprehend and isolate the fault. The facility maintenance office then reviews the output from the module to ensure the request's consistency, correct any errors, and establish a connection with the model. Following this, they apply some basic rules to a fault network based on the model, constructed using relationships between assets, spaces, and symptoms. These rules eliminate the irrelevant potential faults identified by the request and prioritize the remaining ones for forwarding to the site maintenance team. Upon receiving the work order, the site team assesses the potential faults reported, incorporating their on-site experience into the analysis to inspect each identified fault and address the issue. Once they fix the problem, the site team utilizes the site feedback module to report details such as the nature of the problem, the resolution method employed, the root cause of the problem, and an assessment of whether the information provided by the maintenance office was beneficial in addressing the issue. The facility maintenance office methodically reviews feedback from both occupants and the site maintenance team. They standardize the format and content of work orders and employ tags to connect the relations. The updated work order content, reflecting these improvements, is then shared as feedback with the requesters and site workers entering the work orders. This practice aims to minimize the need for content corrections over time, promoting effective communication and continuous improvement in the work order management process. The office manages the whole work order process. Finally, the maintenance office leverages the data analytics module to examine the distribution of work orders, asset replacements, partial changes, and the quality of feedback. This analysis involves statistical and visual assessments, allowing the office to detect potential issues from a comprehensive perspective and proactively resolve them. The details of each module are explained as follows:

Work request module:

The module was designed with the intention of systematically guiding the requester to retrieve all pertinent information about the issue. This systematic approach aims to expedite the detection and maintenance process by ensuring that valuable details are captured efficiently.

It is assumed that comprehensive information about the requester, including name, contact details (e-mail and phone number), occupancy information (room, department, level, building), and any existing maintenance contract, is recorded in the database and updated as needed. As she accesses the request module, potential spaces for reporting complaints are automatically presented according to her identity. For example, a resident is limited to reporting issues within her residence, the common space on the same level, the entrance of the residence, and the parking lot floor. Conversely, a healthcare worker is accountable for addressing concerns only within her specialized department in the hospital. As illustrated in the flowchart of the work request process in Figure 4.1, upon selecting the space, she chooses the symptom(s) of the problems from the dropdown list. Hence, the spatial relations are automatically identified to detect the assets in or intersecting the specified space and then filter them according to the encounters with the reported symptom(s). After that, she can directly select the relevant asset if it is known; otherwise, all potentially faulty assets are observed. In the next session, she employs sensory observations to collect supplementary information on these assets, leveraging the five senses while adhering to maintenance safety regulations. These observations encompass visually inspecting the asset, identifying its sound and direction, physically touching it to identify mainly thermal and pressure-related faults, and detecting any odors arising from the asset. However, tasting the asset's output primarily conflicts with safety

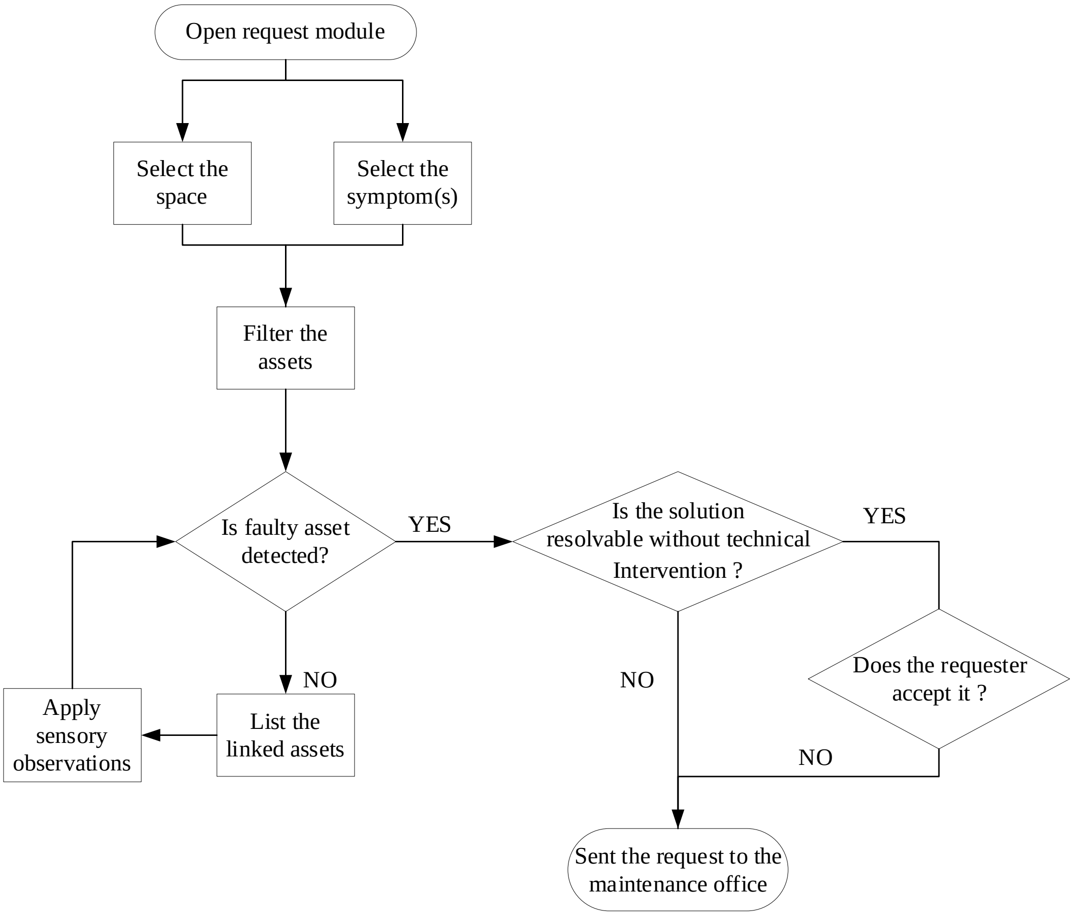


Figure 4.1. Flowchart of work order request

regulations. Visual inspection and tactile examination are the primary senses employed to identify and isolate faults in this session. Whether it's a single asset or multiple assets, the investigation involves exploring potential issues detectable through sensory observations. Depending on the requester's feedback, the module prompts further observation of additional asset(s). For example, suppose a malfunction in the heater is detected through tactile examination. In that case, the module might suggest checking another heater connected to the same system to isolate the location of the problem. This iterative process continues until a final decision is reached. Finally, when the module accurately identifies the problem, and she can address it without needing technical intervention from the maintenance team, it proposes a solution for resolution. If the requester accepts the suggested solution, no further action is necessary. However, if the proposal is declined, the site team is directed to the reported space for further investigation and resolution. At the end of this module, the description of the faults is automatically filled by integrating space, symptom, and asset information with smart tags to connect them to the model. On the other hand, an additional note attribute is provided for the requester to convey any available supplementary information textually. The proposed solution offers ideal steps to maximize the information gain from the requester; however, in practice, some requesters might not be willing to follow all iterative steps. In this case, following the first sensory observations, the requester completes the process unless additional notes are reported.

In addition to the details about the requester, space, symptom, asset, and fault description, the module also incorporates attributes for the occurrence and reported time of the fault, its urgency level, and includes additional documentation and photo attributes. During the hierarchical definition of the problem, asset and space information is reinforced with their 2D-floor view to enhance visual comprehension. Furthermore, visual guidance is extended during sensory observations to streamline the controls.

As outlined in the previous chapter, the faults or suspicious faulty patterns detected in a data-driven environment are integrated into our framework to create work order requests. Once the model is confident about the faulty condition, the data-driven environment is designated as the requester, and the predictive model directly estimates the problem. Hence, the space and asset of the detected problem by the model can be directly extracted from the BIM model since the position of each asset in the system was clearly defined in the BIM model, which has interacted with the sensors previously. Moreover, if the symptom of the encountered fault is not determined beforehand, the symptom attribute is left blank; otherwise, the predefined one is directly assigned to the attribute. In the description, the design inputs and the fault condition are reported. On the other hand, in suspicious patterns, the predictive model struggles to distinguish at least two faulty/normal conditions. Therefore, it presents the successive predictions to the view of the maintenance office to decide how to evaluate the situation. Hence, since at least two different conditions are reported, space, asset, and symptom information cannot be provided unless they are common. Therefore, the description is utilized to both report the selection probability of the successive predictions and explain space, asset, and fault type information of each faulty case with smart tags to connect them with the BIM model.

Work order evaluation and management module:

The case study outlined in the preceding section demonstrates that, despite addressing the same issue, there is a considerable variation in the content and format of work orders when different workers complete the descriptions. Therefore, standardizing both format and content becomes essential, enabling the facility maintenance office to acquire more valuable and consistent information. While the work order request description is automatically generated according to feedback provided by the requester, only the textual additional notes might show variation. Therefore, the facility maintenance office controls the provided information to identify and rectify any contradictory situations, ensuring that additional information adheres to the accepted format. Moreover, when the additional notes encompass intelligent details regarding spaces, assets, fault types, and symptoms, they are incorporated into the model using smart tags to capture model-based relations. In the event of modifications to the request content, the requester is notified, allowing for adjustments in their explanation based on the feedback to enhance the accuracy of future requests.

Following the request update, a fault network analysis approach was developed to apply some filters based on similarities for spatial information, fault symptoms, and the results of existing fault measures. This process aims to isolate the faulty asset and streamline the identification of the root cause of the fault. In cases where multiple potential faults persist, the prioritization of fault inspections is determined by factors related to asset characteristics, maintenance issues, and historical records. Comprehensive details on the network analysis and the factors influencing prioritization are provided in Section 4.5.

After prioritizing the potential root faults, the maintenance office schedules the task and assigns the work order to the on-site worker by transmitting a list containing the work order description and pertinent maintenance information, all coordinated with the BIM model. Hence, the office tracks the status of the work order until it is completed. Once the issue is resolved, the worker reports what the fault was, how it was solved, the action taken, the root of the fault, and the status of the work order is set to “Waiting for Approval.” While the last two were selected from a predefined list or new items were added to the structured list, textual information was provided for the solution description. The office checks the feedback of the site, corrects the format of the description, analyzes the consistency between description, action, and the root cause of the fault, and updates the predefined list if a new item is identified. If the finalized content of the work order is acceptable, its status is marked as “Completed.” In contrast, if the solution description is either incomplete or insufficient for generating valuable feedback, the work order status is revised to "Reporting." It is then returned to the assigned worker with a request to provide a clearer update. For instance, a description like “the desk has been fixed" provides rather general information. On the contrary, a detailed explanation such as "the desk was unstable due to shaky right legs caused by joint failure, and we replaced the joints to address the stability problem" provides a clear summary of the issue and clarifies how joint failures resulted in instability for the desk. Following the updates, the maintenance office links the description, the action taken, and the root cause with the model and relevant assets through smart tags. Additional tags are then defined as keywords to clarify the specificity of the work order compared to others. Hence, the model-based work order is completed.

The textual descriptions and additional notes provided by the requester and the on-site workers must meet the following criteria:

(i) It should be accurate, clear, easily understandable, concise, and consistent.

(ii) It should furnish additional information that enhances the maintenance process.

(iii) It should incorporate information reinforcing the reasoning behind faults, particularly through site feedback.

(iv) The text content should be aligned with the tagging requirements, including smart tags linking the model with content for improving the understandability of the work order and fault network construction systematically, and keyword tags to make the process easier to search, organize, and retrieve relevant information.

In this study, we employed color coding and symbols to establish connections among stakeholders in the fault network analysis. The meanings of the color codes and symbols, along with their corresponding explanations, are detailed in Tables 4.2 and 4.3, respectively.

Table 4.2 Color codes for description tagging

Color	Explanation
Brown	Actions taken to resolve the issue
Green	The cause of the fault utilized for the fault type and asset for intermediate or final cause
Grey	Flow or item flows
Navy blue	Space (linking to the model)
Orange	System (linking to the model)
Purple	The root phase of the fault utilized for the reason an asset
Red	Symptom

Table 4.3 Symbols for description tagging

Symbol	Explanation
	The root reason of the fault
	Asset (linking to the model)
→	Final cause
⟺	Intermediate cause (reason for a fault and cause of another fault)
#	Unmodelled subcomponent of an asset or a system
**	Keyword tag

Site feedback module:

This module is designed to gather feedback on the completed maintenance works from the maintenance site. It aims to enhance the usability of the historical work order records to facilitate the understandability of the future work order request. Following the assignment of the work order, the on-site worker assesses the priority list of fault types, including the fault checklist reported in the request module, based on the site experience and asset criticality. The worker then inspects the relevant cases to identify the root cause of the problem and proceeds with the necessary fixes. When multiple workers are assigned to a work order due to diverse expertise required for fault types listed in the priority list, child work ID is defined to collect each inspection under the same umbrella. Hence, each worker performs tailored inspections specific to the assigned fault types to address the issue. However, in the end, these individual assessments are collectively evaluated to generate a cohesive feedback report for the entire work order. Similarly, when distinct work orders concurrently tackle the same issue, a unified approach is adopted by assessing information gathered from the work requests from these orders together. Moreover, a comprehensive feedback report is generated from the site, utilizing the parent work ID attribute to consolidate the information. Upon finishing the maintenance task, the on-site worker utilizes the module to report the following information to the maintenance office:

(i) *Unique identifiers* to match the site feedback with the work order: work order ID, parent worker order, and child work order ID if needed,

(ii) *Actual start and finish date* and time of the work, whereas *actual duration* is automatically calculated,

(iii) *Description of the solution* including what the problem was, how it was solved, explanation patterns starting from the symptoms and observation throughout the cause and root of the problem,

(iv) *Action* taken and corresponding asset in a structured format,

(v) *Root of the problem* and corresponding asset,

(vi) *Observability* of the detected fault for giving feedback to determine whether a new case can be added to the sensory observations in the work request module.

(vii) *The usefulness of the work description* sent by the maintenance office to isolate the fault and find the root of the fault.

The initial three information entries are directly extracted from the examined tools, while (iv) and (v) are derived from these tools and reformatted structurally, and the final two entries are uniquely defined within this framework. The structural information of (iv) and (v) facilitates the construction of the fault network, whereas (vi) and (vii) provide feedback to enhance the capability of the whole framework.

Data analytics module:

We developed a data analytics module designed to extract work orders generated from the database through model-based processes. This module then analyzes the outputs to assess the efficiency of the constructed framework. The evaluation encompasses (i) the effectiveness of the feedback provided by requesters and the site team and (ii) statistics related to symptoms, spatial aspects, and asset-specific work orders and their correlations with feedback and accuracy.

The module empowers the maintenance office to proactively identify potential issues by inferring insights from work order statistics and descriptions. Regular analysis enables the detection of possible problems and identifies areas for improvement in total accuracy and efficiency within the framework. Additionally, the module interacts with the Building Information Modeling (BIM) environment and may provide a visual representation of the facility's maintenance history. This comprehensive approach facilitates ongoing improvements and a deeper understanding of the facility's maintenance needs.

4.5 A model-based fault network analysis approach for fault reasoning

Inspired by the dynamics of the social network, we proposed a fault network analysis approach to manage the maintenance process efficiently, using the work orders generated in the model-based work order management system. The ultimate aim of the network is to guide the maintenance office in isolating the requested fault and finding its root cause, if possible.

The network, presented in Figure 4.2, is established based on the following relationships:

(i) *Asset vs fault type*: An asset, acting as a fault source, is prone to experiencing multiple types of faults.
(ii) *Symptom vs fault type*: When a fault type occurs in an asset, symptoms arise.
(iii) *Fault type vs fault measure*: Observable signal, specific rule, or system evaluation is followed to isolate the fault type from the others.
(iv) *Root vs cause*: A fault type encountered in an asset causes another fault type of the impacted asset.

In the network, the process flow is represented by arrows, illustrating how a change in one element affects another. This flow originates from a fault type in an asset, leading directly to the emergence of its symptoms, influencing other fault types in different assets if they interact, and ultimately resulting in symptoms across all impacted fault types. Considering that various flows lead to identical symptoms, we backtrack the flow to discard inconsistent alternatives that do not adhere to the specified spatial and observational constraints outlined in the work order request. Moreover, whether the relevant fault type is part of the fault network flow depends on the information obtained from the available fault measure for that specific fault type. If the measure addresses the faulty condition, the flow pathways that do not pass through the fault type are eliminated; otherwise, only these pathways are considered.

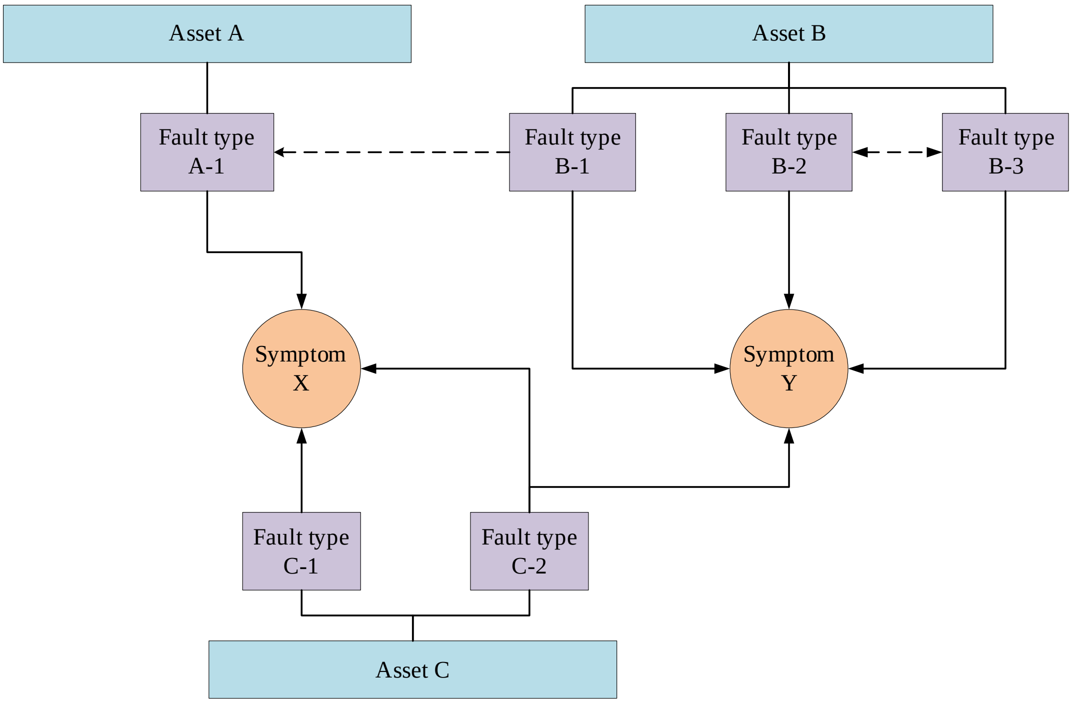


Figure 4.2. Symbolic representation of fault network relations

This network analysis filters the potential fault sources according to the feedback provided by the work order request and the fault measures. As a result of this, the remaining ones are evaluated by the facility maintenance office to report them in priority to the site maintenance team for accelerating the maintenance tasks. The implementation steps of the fault network analysis are explained as follows:

Step 0: Construct a model-based work order management system, identify existing fault types, and propose a checklist for observable faults in each asset. Using this information, develop the fault network.

Step 1: Review the simultaneously filled work order request report and filter the network based on reported symptoms and spatial information.

Step 2: Use the requester's sensory observations checklist to eliminate irrelevant potential fault locations and isolate the network with potential fault types.

Step 3: Analyze the fault measures of these fault types, if available, and eliminate pathways that contradict the results.

Step 4: Prioritize the remaining fault types of the assets according to their criticality and other metrics based on historical work records.

Step 5: Report the prior fault types, their fault source, and location extracted from the model to the site maintenance team.

Step 6: Analyze feedback from the site in the work order report. Update the fault network and observation checklist with new information and adjustments to prioritization. Return to step 1 for a new work order request.

When multiple fault types may persist following fault network analysis, a prioritization method becomes necessary for the site maintenance team to sequence fault inspections. The factors outlined in Table 4.4, based on both practice and literature review, were identified for utilization. However, due to the limited quantity and diversity of available work orders, establishing a widely applicable prioritization method proved challenging. As an alternative, asset criticality is assumed to be the primary factor for prioritization. In cases where fault types possess equal criticality, frequency of occurrence serves as the secondary determinant.

Table 4.4 Factors determining the prioritization of the fault type inspection

Factors	Explanation
Criticality	Prioritize inspecting the fault associated with the asset, which is more critical to the system's functionality or safety and performance.
Interconnectedness	Prioritize the inspection of the asset whose proper functioning is crucial for the performance of other interconnected systems.
Safety concerns	Prioritize the inspection of the fault associated with the asset that presents the higher safety risk if either a fault poses safety risks or has the potential for severe consequences,
Frequency of occurrences	Prioritize inspecting the more frequently occurring fault first and consider the frequency of the faults encountered in the same asset located in different spaces, similar types of assets, and assets with the same functionality.
Similarity with recent work orders	Prioritize the fault complements with the recent work orders.
Ease of inspection	If a fault can be quickly and easily inspected, it might make sense to address it first, especially if time is a critical factor.
Maintenance cost	If one fault, when left unaddressed, could lead to higher repair or replacement costs, it may be prioritized to prevent increased expenses.
Regulatory compliance	Prioritize the faults that could lead to regulatory compliance.
Downtime and repair time	Prioritize the fault to minimize disruption

4.6 Validation case studies

In this section, we verify the procedure employed in the request and site feedback module for generating work orders. Furthermore, the effectiveness and practicality of the proposed model-based work order management system in fault network analysis are substantiated through mini-case studies. These studies demonstrate how the model utilizes smart information to establish relationships, enabling the isolation of fault sources and identification of the root cause of the problem. Four different case studies are implemented, each specifically highlighting a specific capability of the framework.

As explained in the preceding section, the request and site feedback modules were developed within the Dynamo environment, interacting seamlessly with Revit and manipulating model repositories. Utilizing the data-shapes package in Dynamo, we introduced an interface. We collaborated with the Dynamo Player embedded in Revit to gather feedback for work orders from both requesters and on-site workers. Importantly, via these interfaces, individuals without prior knowledge of Building Information Modeling (BIM) can effortlessly manage the completion of work order requests or site feedback. Sample-filled examples for both the request and site feedback modules are provided in Appendix E.

4.6.1 Mini case study 1: prominence of systematic work request - a pressure problem in a combined boiler

This hypothetical case study is designed to illustrate how the work request module is structured to optimize information extraction from complaints while minimizing the additional efforts required by maintenance workers to address the issues.

In the 5-story headquarters of a construction company, the heating system utilizes radiator heaters that operate by circulating hot water to warm the spaces. The heat is generated by a combined boiler situated on each floor. These story-based boilers heat the water and convey it through a network of pipes to the respective radiators on that floor. The radiators, in turn, release the generated heat into the spaces, effectively warming up the spaces.

On a cold winter day, while Engineer A was working in her enclosed office A-307, she became aware that the office was not sufficiently warm for comfortable working. Following this, she inspected potential openings through which cold air could enter from the outdoors. After that, she decided to submit a work order request. Upon accessing the request module, office A-307 was initially assigned to spaces in her request. She then specifically chose the symptom "too cold." In the historical records of the office, this symptom was previously associated only with the air conditioner. However, it was also identified when the radiator was not heating or not heating enough, and heat was escaping from the corners of curtain-walled office windows in various offices throughout the building. Therefore, by employing feature-based relations and space filtering, potential fault sources were narrowed down to three assets: air conditioner, radiator, and curtain wall. In the next step, the request module presented potential detectable faults by sensory observations. Since the air conditioner was not in operation during the working day, she logically excluded this option and proceeded to examine potential faults in other assets. While conducting a tactile examination of the radiator, she visually inspected the curtain wall. Finally, she detected the fault in the office: “The radiator is not producing adequate heat.” After that, the module identified the heating system to which the radiator belongs and provided another instruction based on her spatial accessibility: "Control the nearest radiator in the corridor.", pinpointing the position of the radiator visually in the floor view. This was done to ascertain whether the fault was specific to the radiator in the office or related to another asset in the central system. However, she detected the same problem in the radiator and reported it to the module. Hence, the fault is rooted in the central system, either from the combined boiler or the pipes. In the next step, the module reported the potential observable faults of the combined boiler and pipes. As illustrated in Figure 4.3 for the combined boiler, the fault checklist for each asset is enriched with visual representation to enhance understanding and facilitate the followability of the process. While reviewing the fault list, she identified fault (7) as the pressure in the boiler was significantly low. Hence, the description of the request is defined as follows:

“ The quite lower pressure of the combined boiler significantly decreases the boiler's heat output, causes the radiators, including the ones in the corridor and office A-307, → not producing adequate heat. Hence, the office is too cold.”

After that, the module provided instructions on how to increase the pressure. Once she resolved the pressure issue, the request was logged as statistics and additional information for the regular inspection of the boiler. In the case of more technical problems being reported, the maintenance office would always direct a maintenance worker to address the issue.

	☐ 1. The valves intersect the pipes.
	☐ 2. A special symbol is available in the digital indicator.
	☐ 3. Digital indicators are not working while the control fuse is active.
	…..
	…
	☐ 7. The pressure in the boiler is quite low in the red level.
	☐ 8. The pressure in the boiler is quite high, and water is discharged from the boiler.
	…

Figure 4.3. Visual representation of fault checklist for the combined boiler

This case study verifies the implementation steps of the request module to create a work order request. Moreover, it shows effectiveness in systematically guiding requesters to maximize information retrieval. The module is designed to isolate issues and identify their root causes, thereby expediting the maintenance process. Additionally, in cases involving observable and easily rectifiable conditions that do not compromise safety regulations, the requesters receive instructions to address the issues independently, without the need for intervention from maintenance professionals.

4.6.2 Mini case study 2: model integrated work order management in data-driven facility operations – a stuck coil valve problem in AHU operations

This case study investigates the reporting of a faulty condition in data-driven facility operations within a model-based work order management framework. The analysis explores two distinct scenarios for AHU operations of the second case study in Chapter 3: one addressing a specific confined faulty condition and the other revolving around a suspicious pattern identified by the hybrid intelligence framework.

In the first case, while the Air Handling Unit (AHU) was in operation, the model systematically identified a leakage issue in the heating coil valve. The decision support model confirmed the presence of a leakage problem within the system and seamlessly interacted with the request module to notify the problem to the facility maintenance office. In this case, the AHU system's name was automatically designated as the requester, and the mechanical room associated with the AHU was assigned as the relevant space. As the issue was evident, leaking symptoms are predefined. Consequently, the root cause of the system's problem was documented in the description section, as tabulated in Table 4.5, with the attachment of operational input data.

Table 4.5 Requested work order of leaking heating coil valve

Requester	AHU-2
Space	Mechanical Room-2
Symptom	Leaking
Description	The heating coil valve of AHU-2 system located in Mechanical Room-2 is leaking.

After that, the maintenance office received, accepted, and assigned an HVAC technician to inspect the request. Following the resolution of the issue, the technician reported the solution using the site feedback module. Initially, the solution description stated, "The valve is replaced," lacking sufficient detail. Therefore, additional, comprehensive information was requested to finalize the work order. Responding to the feedback, the technician enhanced the solution description, providing more detailed information as outlined in Table 4.6. The action taken to resolve the issue and its root cause was selected from the dropdown list, with additional information added if necessary. The description offered valuable insights for the root-cause analysis of the solution, marked with "No" for observability, as it required technical knowledge that could not be easily detected by sensory observations. The accurate transmission of the problem to the site technicians facilitated useful feedback. Therefore, this work order report not only confirmed the accuracy of the model predictions but also underscored the crucial role of site feedback in updating the hybrid intelligence model.

Table 4.6 Filled site feedback module for leaking heating coil valve

Work Order ID	AHU-213
Reported By	Ali Asaf
Actual Start Time	2017-09-11 13:14
Actual Finish Time	2017-09-11 13:42
Description of the solution	🔍 Corrosion weakens the structure of 🛠 the valve, leading to cracks. It causes → leakage in 🛠 the heating coil valve of the AHU-2 system located in Mechanical Room-2. 🛠 The valve is replaced.
Action taken	Replacement of the heating coil valve
The root of the problem	Corrosion in heating coil valve
Observability	No
The usefulness of work description	Yes

In the second case study, the decision support model detected suspicious patterns in the conditional prediction of the AHU operations and reported it to the maintenance office. Since at least two different types of fault or fault-free instances exist in the suspicious patterns, only the common attributes are reported. The operational input data is attached to the request. The request details are as follows in Table 4.7.:

Table 4.7 Requested work order report for suspicious fault pattern

Requester	AHU-2
Space	Mechanical Room-2
Symptom	-
Description	2017-09-18 13:15 " No fault" with a probability of 0.85 2017-09-18 13:20 " Stuck outdoor damper in a fully open position" with a probability of 0.72 2017-09-18 13:25 " No fault" with a probability of 0.65 2017-09-18 13:30 " Stuck outdoor damper in a fully open position" with a probability of 0.90 2017-09-18 13:35 " No fault" with a probability of 0.84 2017-09-18 13:40 " Stuck outdoor damper in a fully open position" with a probability of 0.76 A suspicious pattern is detected with respect to two conditions: (1) No fault (2) Stuck outdoor damper in a full open position of AHU-2 system in Mechanical Room 2

Simultaneously, another work order was requested by the occupant of the space where AHU-2 serves; however, in this request, only the symptoms and the fault asset can be identified by the requester, as reported in Table 4.8.

Table 4.8 Requested work order report for the faults in air conditioner.

Requester	Kemal Teke
Space	Test Cell X3A
Symptom	Too cold, Excessive airflow intake
Potential faulty asset	Air conditioner
Description	Excessive airflow is coming from the air conditioner. Therefore, Test Cell X3A is too cold.

The facility maintenance office independently accepted two requests. Despite the suspicious pattern in the data-driven environment, there was an awareness that a stuck outdoor air damper might lead to the same issue reported in the second request. Consequently, both requests were evaluated jointly, and an experienced HVAC technician was assigned to inspect the outdoor damper. The technician was tasked with taking appropriate action to address the issue if identified. After conducting an inspection in compliance with safety regulations by powering off the Air Handling Unit (AHU) unit, the technician identified a break in the wiring between the actuator and the outdoor damper. Despite the actuator sending a signal to the damper, the disconnection resulted in no change in the damper's position. To ensure operational continuity, the technician replaced the broken wire, reestablishing the connection between the actuator and the outdoor damper. After that, the technician reports the findings via the site feedback module, as outlined in Table 4.9. The discrepancies between the predictive model's estimations and the actual faults were evident, as the second request and site feedback confirmed the presence of faults. Consequently, instances predicting "no faults" were stored in a database for inclusion in training sets when updating the prediction model within the hybrid intelligence framework.

This case study verifies the implementation of data-driven work order requests in this framework. Additionally, it integrates the faults predicted in the data-driven environment into model-based work order management. As a result, both data-based and textual recordings interact synergistically with the model, enhancing the generation of intelligent work orders. This integration aims to facilitate the utilization of order intelligence for future requests, contributing to a more efficient and informed work order management process.

Table 4.9 Filled site feedback module report for stuck outdoor damper

Work Order ID	AHU-241
Reported By	Veli Kan
Actual Start Time	2017-09-18 14:21
Actual Finish Time	2017-09-18 16:15
Description of the solution	The broken #wire cuts the connection between actuator and outdoor damper of AHU-2 system in Mechanical Room 2. It causes → stuck in the damper. #The wire is replaced, and a connection is provided.
Action taken	Replacement of wires
The root of the problem	Broken wiring between the control system and the damper
Observability	No
The usefulness of work description	Yes

4.6.3 Mini case study 3: investigation of spatial relations in fault isolation - water leakage problem

This case study aims to demonstrate how spatial relations, specifically those pertaining to space-to-space connections, are assessed in the context of work order generation. In an airport restroom, a passenger noticed water stains, discoloration, or visible wet spots on the ceiling and brought the situation to the attention of the airport staff. Following that, the staff inspected the restroom, closed the relevant sections for use, and accessed the request module to submit the complaints by specifying the location of the restroom and indicating water leakage as the symptom. Upon selecting these options, the module prompted a question regarding whether the issue is detected on the ceilings and their connected walls, aiming to ascertain whether the problem arises from the interior assets or not. When choosing the ceiling, the assets associated with the ceilings of the restroom and the floor above were evaluated, incorporating the projection of the floor onto the ceiling. It became evident that the leakage was located at the projection of the intersection where tee elbow connects two pipes. Therefore, among the checklist reporting the faults related to assets of the wastewater system and toilet fixtures, he opted for the elbow and ticked off "the water leaks in the elbow connections." Hence, the request was reported as outlined in Table 4.10.

Table 4.10 Requested work order report for water leakage issue in the restroom.

Requester	Mehmet Kemal
Space	Restroom 105
Symptom	Water leakage
Potential faulty asset	Elbow tee 502
Description	The water leaks in the # connections of elbow tee 502 of wastewater system 103 in Restroom 205

When the facility maintenance office received the request, they assigned a plumber to fix the problem immediately. The plumber arrived at the maintenance site equipped with a spare identical elbow tee. Using the device to identify water droplets at the reported coordinates of the elbow tee, he successfully pinpointed the source of the leakage, same as reported in the request. After that, he restricted access to the restroom partially, and without delay, he removed all materials obstructing the path to the elbow tee. Upon accessing the problematic area, he observed that the connection of the elbow tee with a pipe was slightly cracked, causing water to leak from the damaged surface. After that, he replaced the cracked elbow tee with the one he had brought and returned everything to its previous condition. Finally, he reported the work order via the site feedback module. The description of the solution was reported in the final work order as follows:

"The crack in # the connection of elbow tee 502 of wastewater system 103 and the pipe 142 causes water leakage, and it results in → water stains, discoloration, or visible wet spots on the ceiling of Restroom 105."

This case study verifies how the projection information can be utilized in fault isolation using space-to-space relations. Additionally, the feedback provided by the requester, with the assistance of a model-based environment, informs the plumber about potential faults and enhances his awareness to take necessary precautions. Therefore, by having the spare elbow tee during the initial inspection, the maintenance task was shortened by eliminating the time spent to identify potential faults part and retrieving new one from inventory, allowing the restrooms to be reopened earlier for use. This not only reduces the potential idle time for the plumber but also minimizes passenger complaints about restroom accessibility.

4.6.4 Mini case study 4: data analytics on the resolution of repeating lighting faults

As explained in the module, data analytics is utilized in work order management to extract valuable insights, identify patterns, and optimize processes, enhancing efficiency and decision-making. In this case study, we examined two hypothetical scenarios inspired by the challenging work orders actually encountered at an airport: (i) a bottleneck in carpentry works and (ii) a rise in lighting faults and associated complaints in a designated baggage claim area.

Carpentry works encompass a range of tasks, including the maintenance, repair, and installation of wooden fixtures such as seating, counters, and custom millwork at the airport. Carpenters play a crucial role in ensuring the structural integrity, safety, and visual appeal of various airport facilities, including waiting areas, lounges, boarding areas, and information desks. Their responsibilities might also extend to flooring, cabinetry, and other wooden elements, contributing to a well-maintained and welcoming environment for passengers and staff. However, the frequent delays in completing these tasks increase complaints. This is due to the restricted availability of seating and the lack of clear signage to assist passengers, as well as occasional noise and disruption. Additionally, access to amenities like restrooms and information desks is often restricted. This problem sometimes results in delays in boarding and departure. Therefore, the facility maintenance office regularly analyzes performance metrics of maintenance works including the carpentry works to improve maintenance efficiency and prevent potential complaints. In one of these analyses, they first investigated the performance of the carpenters; however, no significant differences were observed in their performance. Then, a detailed analysis was undertaken to assess the space-based performance of the carpentry works, specifically the works conducted in the waiting and boarding areas. In this analysis, the works on the second floor and those on the basement floor are compared. Despite the works being similarly distributed across both floors, the average execution time of tasks on the second floor was found to be reasonably higher than those on the basement floor. To delve deeper into the analysis, the statistics of other attributes were examined, and a significant clue was captured in "the usefulness of work description." While the carpenters found most of the work description informative and value-additive for the basement floor, this rate decreased to two-thirds on the second floor. The carpenters also validated this finding. Given the extensive scope of the work, the feedback provided by the requester emerged as a pivotal factor in streamlining the maintenance process and expediting the tasks. Therefore, a collaborative meeting was arranged to explore ways of enhancing requester feedback. After that, a training program was formulated to educate requesters on how to submit clear and comprehensive work order requests. Through this program, in the next regular inspection six months later, the feedback efficiency of requesters from the second floor for the cases deemed not informative improved in at least one out of the five instances. As a result, the execution time of the carpentry works for that floor was considerably improved. It demonstrates how initial efficient feedback on requested work significantly enhances the entire maintenance process, especially concerning carpentry works.

The complaints of the passengers and airport staff are systematically collected to enhance the operational efficiency of the airport. These concerns are conveyed to the corresponding management units regularly to address and resolve the issues. The facility maintenance office reviews these complaints as part of its routine analysis of work orders. Some of these complaints specifically mention the lighting issue in the baggage claim area 4, citing discomfort in the visual experience due to excessive brightness. Upon examining the statistics of lighting faults in baggage claim areas, it was observed that despite the use of the same light bulbs in other areas, certain bulbs in the pinpointed area were replaced three times within a short period.

While the same cable line powered all the replaced bulbs, other lines connecting to the same electrical panel did not encounter any faulty issues. The historical records indicate that the maintenance task was previously performed on the same panel before these problems arose. This issue is likely a result of human error during or after that task, pointing to a problem in the connection between the panel and the cable line. The reported excessive brightness in complaints directly highlights a configuration problem in the connections of the cable line and electric panel. Hence, a work order was initiated, assigning an electrician to inspect the current condition and address any identified issues. According to the report, the electrician detected and resolved the configuration problem. In the next routine analysis, no significant complaints on the same issue were encountered.

4.7 Discussion and conclusion

In this research, we developed a model-based work order management framework that links assets and spaces to the corresponding ones in the BIM model. The observable symptoms, the source asset of the fault, spatial information, and the impacted assets are symbolized with tags, including color coding. The linked models and tagged order are brought together to detect spatial, systemic, and feature-based relations automatically and make the work order smarter, addressing RQ 3a. Hence, information gained from these linked historical records is utilized to facilitate fault isolation and reasoning. The content of the framework is limited to the four modules: request, site feedback, evaluation and management, and data analytics. The request module was designed in a guided structural form to maximize the information gained initially for detecting the problem accurately and directing the maintenance team with enough information. The site feedback module was proposed to collect constructive feedback on the completed maintenance works. The evaluation and management module was constructed to control, correct and link the work requested and work order completed. In other words, it is the control points managed by the facility maintenance office to standardize the entries of the work order forms provided by different requesters and maintenance site workers. Moreover, a fault network analysis approach was developed, addressing RQ 3b to construct the relations between fault symptoms, assets, asset-specific fault types, and fault measures if they exist. When a new work is requested, according to the information provided, utilizing the similarities in spatial information, fault symptoms, and the results of existing fault measures, filters are applied to isolate the fault and detect its root. The facility maintenance office evaluates the outputs of the analysis and assigns the corresponding maintenance worker to inspect the fault on site. Finally, a data analytics module was developed to analyze the statistics of the existing work orders regularly to identify potential issues by inferring insights and taking necessary precautions proactively. The applicability of the proposed framework is verified with four different case studies that show how the proposed fault network approach and smart work orders linked with model elements defining the relations automatically. Hence, by employing fault network analysis alongside data obtained from work requests, the framework's ability to isolate faults and potentially identify their reasons is validated. As a result, this framework streamlines fault isolation and reasoning processes through the utilization of model-based relations, facilitating a model-driven approach.

From the point of information needs, first, space and symptom information is received, and then, using asset-space relations, the potential faulty assets are listed. To isolate the fault, fault lists based on sensory observations of the assets are required to be provided to the requester for checking. Additionally, when the resolution of the problem is non-technical, information on how to resolve it is needed to guide the requester. In the site feedback module, while actions taken to handle the issue and its root are received in the structural form, observability of the resolved issue and usefulness of the work description sent to the site workers are evaluated to give feedback for updating the framework to improve prediction accuracy.

The facility maintenance office standardizes the format and content of work requests and solution descriptions within the framework. While the description for both semantic and data-driven work requests is automatically generated from the guided structural form, and rules are defined for entries in the site feedback module. If the description lacks sufficient details to explain the completed work, feedback is sent to the responsible worker for a more detailed redescribing of the task. Following the completion of the request and work order descriptions, they are linked to the corresponding model elements. When ordinary people read the color-coded and tagged descriptions, they can easily construct fault patterns using the established relationships.

In the development of the framework, detailed information is required about potential faults of the assets available in the model to facilitate the isolation of the issues in the request where the faults can be observed by five senses. Therefore, in the handover stage, a guideline and form should be prepared to standardize the fault information collection and gather them from the vendors, including the asset, its subcomponent, its corresponding fault type, symptom(s), and fault information, and action taken to handle the issue. The information that is not available for the assets, especially for structural and architectural components, needs to be collected before the handover stage. Since it provides system-specific or asset-specific fault information, including symptoms, fault types, and their relations, the historical records of similar facilities can be utilized to adapt their information to the new one. On the other hand, information collected for newly defined assets or systems can be integrated into the fault network in a customizable manner. Moreover, the linkage of the assets with the work order depends on the availability of assets or their subcomponent in the model. If it is modeled, the work order is linked to the asset; otherwise, the "#" symbol is presented in the description to link it to the most relevant asset or system. Furthermore, in the application of the framework, when a new fault case that is not defined before emerges, information is collected via additional notes from the request module, and the case is inspected and resolved by the site workers and reported via the feedback module. In the case of new information emergence for the structural information presented in a dropdown list or checklist, including symptoms or action taken to handle the issue, it is allowed to add new items and then controlled and corrected by the facility maintenance office. When a new observable fault is detected by the site workers via the observability attribute, feedback is given to the maintenance office. The fault checklist of the asset by sensor observations is then updated.

In the future, as the usage of the framework matures over time, the responsibilities of the facility maintenance office in controlling and correcting the content of work orders and requests will gradually be delegated to an AI model. This transition will occur once a sufficient amount of work order data is collected to train the model. Hence, it will provide decision support for the maintenance office to accelerate the process. Moreover, the office controls the model outputs, corrects them, and gives feedback to update the AI model regularly for continuous improvement. In addition, as explained in Section 4.5, the prioritization of fault reasoning when more than one alternative is available will be formulated. Furthermore, since BIM facilitates the linking the relations between assets and spaces and only observable faults are considered in this framework, developing an asset-specific expert system for fault reasoning and integrating it into the framework will broaden its coverage and improve the systematic and comprehensive analysis of faults within the facility.

CHAPTER 5

CONCLUSIONS AND FUTURE RESEARCH

In this research, a conceptual framework for BIM-driven facility maintenance and fault management is developed to address the entire maintenance process by reviewing previous studies and practices. The research's focus begins from the pre-maintenance stage encompassing asset tracking and system monitoring, fault detection and reasoning, work order management, and information management in maintenance tasks. To facilitate information management in this process, the framework concentrates on information requirements, standardization and resolution of interoperability and integration issues in model-based environments, efficient utilization of information technologies to streamline maintenance tasks, and specifically, needs analysis for model-based work order and fault management. To eliminate potential significant time and money lost due to information losses and delays in understanding the real reason for the faults in a facility, the research gaps are identified, and the potential of BIM is leveraged to develop two decision-support solutions for fault reasoning.

In the first solution, considering the need for adaptation to uncover the behavior of the whole system and provision of practical reliability of the monitoring system, a hybrid intelligence approach is developed to interact humans with AI models for consistently improving the fault reasoning throughout the lifecycle of the facility to meet the needs of data-driven fault reasoning and integrating with the BIM environment to link the data-driven solution with maintenance workflow. The approach comprises six interconnected modules. Firstly, the design module constructs the sensor network as design inputs. Secondly, the data acquisition and preprocessing module collects data through real-time monitoring or simulation, cleaning, and sampling it for training and testing purposes. Thirdly, the predictive modeling module selects or designs the appropriate AI algorithm and trains it for fault detection and reasoning. Fourthly, the monitoring and decision support module tests the constructed algorithm with incoming data, monitors consecutive predictions, and reports suspicious patterns as work order requests to the facility maintenance office. The fifth module, the human-centric evaluation, involves the maintenance team in assessing requests, conducting on-site inspections, addressing detected problems, and providing feedback to enhance prediction accuracy. Lastly, the continuous enhancement module utilizes feedback from site inspections and the decision support model to improve data support for fault detection and reasoning continually. BIM is integrated into the approach to interact with the sensor network for extracting relevant sensor information for corresponding assets and with the decision support model to provide spatial and maintenance information and documentation for the work order request. As highlighted above, the interaction between humans and machines is pivotal for enhancing fault reasoning predictability and advancing the sustainability of the approach. Humans are central in various aspects, including network construction development, data preprocessing, predictive model selection, decision-making in fault management, and offering constructive feedback from site inspections to enhance machine predictions and improve prediction accuracy. In contrast, AI models streamline repetitive and time-consuming processes within the modules and provide invaluable decision support by predicting fault reasons, thereby facilitating human decision-making. Since the predictive model is one of the main determinants of efficient fault reasoning, we implemented hierarchical classification, strategically dividing the extensive array of fault classes into more manageable units. This strategy proves particularly beneficial in systems housing numerous fault types with varying severity levels across assets. The architecture of the algorithm mirrors the fault hierarchy, coordinating multiple local classifiers in hierarchical alignment. It systematically detects the fault's presence, pinpoints the fault's source asset, identifies its fault type, and evaluates its severity level. This structured methodology aids in both understanding and resolving issues efficiently. Furthermore, to tackle errors propagated through the hierarchy, we devised local classifiers utilizing a Back-propagated Neural Network with conditional stepwise learning, thus minimizing the risk of overfitting. Additionally, we trained additional local classifiers to offer constructive feedback for updating the primary hierarchical model. This process involves determining combined contribution weights via linear regression from the feedback classifier(s) and the main model, thereby enhancing the predictive accuracy and robustness of the overall model. As a result, the performance of the proposed approach and AI models is validated with two case studies by comparing its solution quality with the common AI classification models.

In the second solution, a model-based work order management framework that connects assets and spaces with their counterparts in the BIM model is proposed to facilitate information gains while the problem and its solution are reported. Observable symptoms, fault source assets, spatial information, and affected assets are represented using tags, including color codes. These linked models and tagged orders enable the automatic detection of spatial, systemic, and feature-based relationships, thus enhancing work order intelligence. Hence, information gained from these linked historical records is utilized to facilitate fault isolation and reasoning. The framework comprises four modules: request, site feedback, evaluation and management, and data analytics. The request module is structured to maximize initial information acquisition for accurate problem detection and to guide the maintenance team effectively. The site feedback module collects constructive feedback on completed maintenance tasks. The evaluation and management module controls, corrects, and links requested work with completed work orders, serving as control points managed by the facility maintenance office to standardize entries from different requesters and maintenance workers. Furthermore, a fault network analysis approach is proposed to establish relations between fault symptoms, assets, asset-specific fault types, and fault measures, if applicable. When a new work order is requested, filters are applied based on spatial information, fault symptoms, and existing fault measures to isolate and identify the root cause of the fault. The facility maintenance office evaluates the analysis outputs and assigns maintenance workers to inspect the fault on site. Lastly, a data analytics module was created to regularly analyze existing work order statistics, infer insights, and proactively take necessary precautions to identify potential issues. The proposed framework's applicability was verified and validated successfully through four case studies, demonstrating how the fault network approach and smart work orders linked with model elements automatically facilitate fault isolation and reasoning.

5.1 Summary of contributions

This research contributes to several aspects of model-based facility maintenance management and fault reasoning to facilitate the maintenance tasks:

Review-based facility maintenance management framework: Current literature studies focus on the restricted aspects of BIM-based facility maintenance to manage information flow and maintenance workflow. This research introduces a conceptual framework founded on comprehensive reviews of existing practices, methodologies, and technologies in model-based maintenance management. By synthesizing ideas from the reviews about the strengths and weaknesses of current approaches, the framework offers a robust foundation for efficient facility maintenance operations, including information requirements in different stages of the facility lifecycle, utilization of information technologies to facilitate maintenance information workflow, compatibilities for interoperability and integration of BIM with other technologies, and direction for model-based work order and fault management.

First hybrid intelligence application in fault reasoning: The efficacy of current AI classification models for fault detection and reasoning hinges upon the quality, relevance, and representativeness of the available data, which are imperative for demonstrating the holistic behavior of the system. However, throughout the system's life cycle, unforeseen conditions and repetition of similar mispredictions due to the capability of the current AI model diminish the reliability of the proposed solution in practice. Therefore, to address the drawbacks, this study presents a pioneer solution to data-driven fault reasoning, called the hybrid intelligence approach, to provide iterative feedback within the interaction of humans and AI models for consistently improving fault detection and reasoning predictions. While humans construct the data-driven monitoring network, preprocess the initial training data, select and fine-tune the AI model intuitively, provide constructive feedback from site inspection to machine prediction, and finally make a decision on fault conditions in practice, AI models collect the real-time data and makes predictions to provide decision support for human-decision-making. The collected feedback for mispredictions detected by the decision support model and site inspection is utilized to resample the training data and retrain the AI model for regular updates. Hence, by integrating human expertise and insights with the AI model's capabilities, both parties augment each other for continuous improvement in the predictions.

A novel feedback mechanism to improve the performance of classification models: The hierarchical classification model orchestrates multiple independently trained local classifiers through a structured sequence of hierarchical decision processes to finalize its reasoning predictions. However, mispredictions at any level of the hierarchy propagate and accumulate errors. Therefore, in order to regain a significant portion of the information loss through this hierarchy, a novel feedback mechanism is developed to improve the prediction accuracy, especially for conflicting fault classes where mispredictions exceed the predefined threshold values. In this path, examining the confusion matrix of the training data, independent subsets, including conflicting classes, are determined, and an additional local classifier is trained using the dataset of the classes in each subset. After that, least square regression is employed to minimize the square error between actual faults and feedback-enhanced predictions, optimizing the combined weights of the hierarchical main model and additive classifier for each subset. Hence, constructive feedback from the conflicting classes is integrated into the main hierarchical model to improve model accuracy. This feedback mechanism offers a robust alternative to ensemble models, including bagging, boosting, and stacking ones, in the classification problem literature, as evidenced by applied case studies.

A knowledge-assisted model-based fault reasoning: The previous studies established relationships between problem sources and model elements using predefined manually customized templates and case studies, each creating a database for specific situations, from which algorithms were employed to search and filter relevant templates. Hence, a root-cause mechanism associated with selected templates identified problem sources and correlated model elements solely linked to the chosen template, whereas the intelligence of BIM was not actively utilized to define the templates and fault patterns. In contrast, this research is founded on the utilization of model-based relations and the information gained from the relevant historical work order records to isolate and localize the faults. Therefore, a model-driven framework to manage work orders is designed in a Dynamo environment to structure and standardize the content and format of work orders to establish connections among observable symptoms, the source asset of the fault, spatial information, and the impacted assets within the BIM environment. Moreover, smart tags are employed to construct a fault network using the relations between asset and fault type, symptom and fault type, fault type and fault measure, and finally, between fault types defining the root and cause of the problem and connect the work orders to the model elements. Based on the information provided by the maintenance requester on fault space, symptom, asset, and system, the fault network is filtered to isolate the potential fault reason(s) and pinpoint it if possible. Since the faults are defined specific to the assets or systems, while replacing the type of existing assets or constructing fault network for a newly built facility, the general fault network is customizable to meet the specific requirements of the facility.

5.2 Limitations of research

While the contributions of this research mark significant advancements to support facility maintenance, they also present some limitations, as reported below:

- Although the abstracts of all reported articles are examined one by one, the articles collected from Web of Science and Scopus databases to evaluate for literature review of model-based maintenance management and fault reasoning to construct the conceptual framework are restrained with certain keywords such as “BIM” and “maintenance,” “BIM” and “work order,” and “BIM” and “fault detection and diagnosis.” The relevant studies on facility maintenance, work order management, and fault reasoning, but those that are out of BIM scope, have not been evaluated.

- The hybrid intelligence approach offers continuous improvement to detect and reason the faults throughout the lifecycle monitoring of the systems within the facility. Using a real-time dataset on a longer time horizon within regular updates would be more efficient in validating the efficiency of this approach. However, due to the absence of data availability, the datasets generated for fault patterns using experiments and simulations are evaluated in the validation process. Since at least two iterative steps are required to show consistent improvement in fault predictions, the datasets in time series were divided into five sets, as reported in Chapter 3. While the model was first trained with the three sets and tested with the fourth one, in the second step, the samples of the incorrectly classified test instances were added to the training data, and the model was retrained and tested with the fifth set.
- The feedback-enhanced hierarchical classification model in the hybrid intelligence approach is designed assuming that a single fault exists at a time in the system, compatible with validation cases; however, the scenario involving multiple faults detected simultaneously is also discussed in Chapter 3. Nevertheless, in that model, the existence of faults and the classification of their severity levels are modeled as in the original one. However, in contrast to our model, the fault hierarchy, including faulty assets and their fault types, is deconstructed, and the sigmoid function replaces the softmax activation function for multi-class classification and binary classifiers are exchanged with two-class multi-classifiers. While the softmax function allows for relative comparison and addresses a faulty asset and fault type for the continuity of the hierarchical classification, the sigmoid function

evaluates each alternative independently. As a result of this, while a fault can be detected by the fault existence classifier at the top node of the hierarchy, none of the fault types of the assets might theoretically be addressed at the next level of the hierarchy, resulting in information losses and limitations in information transfer.

- The model-based work order framework primarily concentrates on the content and format of work orders and their integration with the BIM model to enhance the intelligence of work orders and utilize acquired information to expedite fault isolation and reasoning. Therefore, certain conventional aspects of work order management systems, such as scheduling and prioritizing work orders, optimizing resource utilization, integrating with other business systems, and related considerations, are not directly covered within this framework. Initially, our intention was to develop an index for prioritizing the remaining fault types if more than one exists following the receipt of a work order request. However, we encountered challenges in establishing a prioritization method that could be widely applicable, primarily due to the limited quantity and diversity of available work orders. Therefore, the criticality of the assets was evaluated to prioritize the inspection order.
- In knowledge-assisted model-based fault reasoning, smart tags or additional tags are utilized to link work order descriptions to the relevant assets, their fault types, space, system, and symptoms in the model. Using the information provided by historical records and model-based relations, potential faults are isolated, and the reason for the fault is pinpointed if available. However, while linking information gained from the work orders, the “garbage in, garbage out” principle comes into play for the model. In cases where the asset and its subcomponents are fully modeled, the relevant ones are easily linked with the work order description. Otherwise, a practical approach involves utilizing placeholders or indicators, such as the "#" symbol, within the work order descriptions. This symbol serves as a reference point, indicating that

the work order is linked to the most relevant asset or system, even if the asset is not fully modeled or specified within the BIM environment. If the boundaries of the spaces and relations of the systems are not clearly defined, it influences the accuracy and reliability of the fault reasoning model.

5.3 Practical implications and recommendations for future studies

The journey of this research has not only uncovered novel findings but has also illuminated practical pathways for improving information management in facility maintenance and streamlining fault reasoning. To begin with the conceptual framework for model-based facility maintenance and fault management, it increases the awareness of the facility managers for lifecycle model-driven maintenance management by drawing the whole picture of the maintenance issues integrated with BIM and information technologies to facilitate the tasks, not only for asset tracking and system monitoring, but also fault detection and diagnosis and maintenance training and tracking. The framework advocates for a proactive role of the facility management office during the design stage of the facility rather than waiting until the handover stage before the operational phase. This proactive involvement ensures that potential issues related to accessibility paths and design are identified and rectified earlier to prevent design faults and repair needs later. Leveraging BIM-integrated virtual solutions, such as simulations and model-based design checks, facilitates this process by providing insights into the facility's functionality and identifying any design-related shortcomings. Additionally, during the handover stage, a detailed list of maintenance items from previous studies and industry practices can be used to create a specialized checklist for maintenance issues. This helps ensure that all necessary maintenance considerations are addressed from the start, setting the stage for smoother operations and upkeep in the long run. Moreover, standardization of data format, naming, and information content facilitates data exchange between maintenance tools.

In addition to universally accepted formats like IFC, IFC-FM, and COBie, this framework also offers alternative, customizable solutions using application programming interfaces (API) and visual programming tools. These tools enable the manipulation and extraction of data from models, allowing for the creation of customized sheets tailored to specific needs. This approach streamlines repetitive tasks during handover and model updates, enhancing efficiency and accuracy. Furthermore, the framework demonstrates how various information technologies such as technologies such as internet of things, augmented, virtual, and mixed reality, GIS, barcodes, especially QR codes, game engines, RFID, blockchain, digital twin, robotics, laser scanning, chatbots, artificial intelligence, and natural language programming, enterprise resource planning are integrated within the BIM environment. These technologies facilitate data collection, information display, retrieval, and management, thereby supporting maintenance tasks effectively. By incorporating these technologies into the BIM environment, facility managers are empowered to establish a robust technological foundation during the planning stages. This foundation facilitates seamless operations from a maintenance perspective, enabling efficient and effective maintenance activities throughout the lifecycle of the facility.

Apart from the supportive concepts within the conceptual framework, two model-integrated decision-support solutions for fault reasoning are highly practical in application. The hybrid intelligence approach is well-suited for continuously monitored systems reliant on sensor-based tracking within data-driven environments. These systems encompass a wide array of facilities, including heating, ventilation, and air conditioning, environmental monitoring, electrical, lighting, plumbing, elevator, and lifting systems, as well as industrial equipment and machinery. On the other hand, knowledge-assisted model-based fault reasoning is highly beneficial for tackling general maintenance issues across various systems. This approach benefits from observable symptoms to effectively distinguish between faults, which directs the requester to maximize the information gained for isolating the faults. While the first approach provides a solution centered around the system itself, the second framework offers a search focused on spatial analysis. However, both solutions are open to improvement through constructive feedback provided for new and misevaluated cases.

To apply the hybrid intelligence approach, the facility maintenance team should be experienced and knowledgeable about the fault patterns of the system. Due to the scarcity of data collected initially from the system, developing predictive models for initial training poses challenges. To overcome these challenges, domain expertise is required to label the faults and develop strategies for initial training data collection such as (i) simulating data to represent expected fault patterns, (ii) leveraging pre-trained models for similar systems and fine-tuning them with synthetic or limited real-world data, (iii) collaborating with domain experts to understand fault behaviors and manually label data, and (iv) testing faults in controlled environments to refine the model before deployment in more complex scenarios iteratively. This approach is more practical for organizations that operate more than multiple facilities with identical functionalities and utilizing the same systems. Hence, lessons learned from the models constructed for the operated facilities can be applied while creating the model of the new one. Moreover, the feedback-enhanced hierarchical classification model, which predicts the existence of the fault and its reason, offers generalizable data-driven fault reasoning solutions. In practice, priorly, the model with multiple fault detection simultaneously, instead of a single fault at a time, is directly applicable to the relevant systems; however, in some cases where the number of fault classes is limited, multi-class classification can be applied with feedback mechanism instead of hierarchical model to reduce the complexity of the problem and computational time. Additionally, specific to the used system, some faults are easily separable from others by defining simple, precise rules; hence, no feedback is required to correct any misprediction. In this application, the predictive model should be constructed using the remaining fault classes after filtering the separable faults. In real-time monitoring, incoming data instances are tested with both rules and a predictive model. While the rule-based fault detections are directly reported as work order requests, the maintenance team defines rules as a decision support model to make a tradeoff between workload to examine faulty data instances predicted and suspicious pattern changes in the consecutive data instances, and misevaluations including missing faults, misclassified faults or false alarms for fault inspections. The rules defined in this research can be initially utilized; however, each system evolves its own set of rules over time in practical application. As humans are the main playmakers of this maintenance approach, especially at the beginning of the newly constructed system, their judgments are critical for ensuring a sound and accurate start in data-driven fault reasoning. While integrating BIM with a hybrid intelligence approach enables consistency and continuity in the maintenance process, the standalone version is also viable for detecting and reasoning faults within the facility's lifecycle.

In contrast to the previous approach, knowledge-assisted model-based fault reasoning relies on BIM-based relations. Therefore, to implement this solution in practice, having a BIM model of the facility is essential. Hence, with respect to the "garbage in, garbage out" principle, the accurateness and efficient applicability of the model-centered solution are directly impacted by the completeness of the model. The geometric boundaries of the spaces and their relations with assets need to be clearly defined. Moreover, comprehensive information is essential about potential faults of the assets available in the model to facilitate the isolation of the issues in the request where the faults can be observed by five senses. Therefore, during the handover stage, it is imperative to prepare guidelines and forms to standardize the collection of fault information. This entails gathering details from vendors, including the asset, its subcomponent, corresponding fault type, symptoms, fault information, and actions taken to address the issue. As it furnishes system-specific or asset-specific fault information, encompassing symptoms, fault types, and their interrelations, historical records from similar operating facilities can be utilized to customize their information for the new facility.

In practice, multiple workers and occupants contribute to the generation of work order content. To ensure consistent generation of historical records, the framework facilitates standardized formats and content for work requests and solution descriptions. Workers are required to describe completed tasks in accordance with predefined rules to standardize explanations. Maintenance requesters are guided through structured forms to generate semantic and data-driven work requests automatically. This guided content of the request, including structural information, helps alleviate the reluctance of occupants to provide detailed information. Similar to the previous approach, the facility maintenance team's efforts are required to manage the work order process and link them with corresponding model elements via color-coded and tagged descriptions; hence, even ordinary people can easily identify fault patterns by utilizing the established relationships.

In addition to the practical implications of our fault reasoning solutions, in future studies, the capabilities of both solutions will be enhanced with more complex scenarios. To begin with the hybrid intelligence approach, as the number of fault classes is relatively limited in the validated case studies, the robustness of the approach will be examined with more complex cases in a living environment encompassing a diverse range of fault types and severity levels across a broader spectrum of assets within a system. Expanding the scope of analysis aims to validate the effectiveness and investigate the reliability of the framework under varied and challenging conditions. This comprehensive examination will provide valuable insights into the adaptability and scalability of the approach to enhance its practical utility in real-world applications. Furthermore, as the framework gains traction and evolves, there will be a gradual transition towards leveraging AI models to streamline the management and correction of work orders and maintenance requests within facility maintenance offices. With the accumulation of a substantial volume of work order data, the AI model will be trained to provide decision support, expediting the processing of maintenance tasks. The maintenance office will retain control over the model outputs, ensuring accuracy and relevance while also actively engaging in the continuous refinement of the AI model through feedback mechanisms. Additionally, efforts will be directed toward formulating a prioritization mechanism for fault reasoning in cases where multiple alternatives exist, enhancing the efficiency of fault management processes. Moreover, the integration of asset-specific expert systems for fault reasoning, alongside the BIM utilization to establish linkages between assets and spaces, will further enrich the analytical capabilities of the framework, enabling a more systematic and comprehensive analysis of facility faults. This integrated approach promises to significantly enhance fault detection and resolution within the facility management domain, ultimately optimizing maintenance operations and enhancing overall operational efficiency.

REFERENCES

Abid, A., Khan, M. T., & Iqbal, J. (2021). A review on fault detection and diagnosis techniques: basics and beyond. *Artificial Intelligence Review*, *54*(5), 3639–3664. https://doi.org/10.1007/s10462-020-09934-2

Akanmu, A. A., Ojelade, A., & Bulbul, T. (2018). Gaming approach to designing for maintainability: a light fixture example. *Proceedings of the 35th ISARC*, 1113–1116. https://doi.org/10.22260/isarc2018/0154

Akanmu, A. A., Olayiwola, J., & Olatunji, O. A. (2020). Automated checking of building component accessibility for maintenance. *Automation in Construction*, *114*. https://doi.org/10.1016/j.autcon.2020.103196

Akcamete, A. (2011). *A formal approach for managing facility change information and capturing change history as part of building information models*. PhD Dissertation, Carnegie Mellon University.

Akcamete, A., Akinci B, & Garrett JH. (2010). Potential utilization of Building Information Models for planning maintenance activities. *Proceedings of the International Conference on Computing in Civil and Building Engineering*. http://www.engineering.nottingham.ac.uk/icccbe/proceedings/pdf/af76.pdf

Alankarage, S., Chileshe, N., Samaraweera, A., Rameezdeen, R., & Edwards, D. J. (2023). Organisational BIM maturity models and their applications: a systematic literature review. *Architectural Engineering and Design Management*, *19*(6), 567–585. https://doi.org/10.1080/17452007.2022.2068496

Alavi, H., & Forcada, N. (2022). User-centric BIM-based framework for HVAC root-cause detection. *Energies*, *15*(10), 3674. https://doi.org/10.3390/en15103674

Ali, A. S., Zakaria, N., & Zolkafli@Zulkifly, U. K. (2021). Building operation and maintenance: a framework for simplified Building Information Modeling (BIM) digital mobile application. *International Journal of Interactive Mobile Technologies*, *15*(20), 146–160. https://doi.org/10.3991/ijim.v15i20.23753

Alizadehsalehi, S., Hadavi, A., & Huang, J. C. (2020). From BIM to extended reality in AEC industry. *Automation in Construction*, *116*, 103254. https://doi.org/10.1016/j.autcon.2020.103254

Alshokry, G. B., Hagal, M. A., & Aljabour, B. A. (2021). Tracking and reporting software maintenance requests challenges in CMMS systems: proposing a custom computerized maintenance management systems CMMS tool, especially designed for software systems maintenance. *The 7th International Conference on Engineering \& MIS 2021*, 43. https://doi.org/10.1145/3492547.3492621

Altun, M. (2015). *Model based building energy optimization using meta-heuristics* [Middle East Technical University]. https://open.metu.edu.tr/handle/11511/24968

Artan, D., Ergen, E., Kula, B., & Guven, G. (2022). Rateworkspace: BIM integrated post-occupancy evaluation system for office buildings. *Journal of Information Technology in Construction*, *27*, 441–485. https://doi.org/10.36680/j.itcon.2022.022

Banerjee, A., & Nayaka, R. R. (2022). A comprehensive overview on BIM-integrated cyber physical system architectures and practices in the architecture, engineering and construction industry. *Construction Innovation*, *22*(4), 727–748. https://doi.org/10.1108/CI-02-2021-0029

Borhani, A., & Dossick, C. S. (2020). *Data Requirements for BIM-Based Asset Management from Owners' Perspective* (T. P., G. D., & E. A. M. (eds.); pp. 478–487). American Society of Civil Engineers (ASCE).

Bortolini, R., Forcada, N., & Macarulla, M. (2016). BIM for the integration of building maintenance management: A case study of a university campus. *Proceedings of the 11th European Conference on Product and Process Modelling*, 427–434. https://doi.org/10.1201/9781315386904-62

Bouabdallaoui, Y., Lafhaj, Z., Yim, P., Ducoulombier, L., & Bennadji, B. (2020). Natural language processing model for managing maintenance requests in buildings. *Buildings*, *10*(9). https://doi.org/10.3390/BUILDINGS10090160

Camus, T., & Moubarak, S. (2015). Maintenance robotics in TBM tunnelling. In R. Heikkilä (Ed.), *Proceedings of the 32nd International Symposium on Automation and Robotics in Construction and Mining* (pp. 1–8). International Association for Automation and Robotics in Construction (IAARC). https://doi.org/10.22260/ISARC2015/0090

Cavka, H. B., Staub-French, S., & Poirier, E. A. (2017a). Developing owner information requirements for BIM-enabled project delivery and asset management. *Automation in Construction*, *83*, 169–183. https://doi.org/10.1016/j.autcon.2017.08.006

Cavka, H. B., Staub-French, S., & Poirier, E. A. (2017b). Developing owner information requirements for BIM-enabled project delivery and asset management. *Automation in Construction*, *83*, 169–183. https://doi.org/10.1016/J.AUTCON.2017.08.006

Cavka, H. B., Staub-French, S., & Pottinger, R. (2013). Case study of BIM handover to support building operations. In K. C.J. & B. C. (Eds.), *Proceedings of Annual Conference - Canadian Society for Civil Engineering* (Vol. 2, Issue January, pp. 1159–1168). Canadian Society for Civil Engineering. https://www.researchgate.net/publication/289744294_Case_study_of_BIM_handover_to_support_building_operations

Chan, P.-S., Chan, H.-Y., & Yuen, P.-H. (2016). BIM-enabled streamlined fault localization with system topology, RFID technology and real-time data acquisition interfaces. *2016 IEEE International Conference on Automation Science and Engineering*, 815–820. https://doi.org/10.1109/COASE.2016.7743486

Chang, J. X., Su, Y. C., & Lin, Y. C. (2013). Development of mobile BIM-assisted defect management system for quality inspection of building projects. *Proceedings of the Thirteenth East Asia-Pacific Conference on Structural Engineering and Construction*, B-3-6. http://hdl.handle.net/2115/54248

Chen, K. L., & Tsai, M. H. (2021). Conversation-based information delivery method for facility management. *Sensors*, *21*(14). https://doi.org/10.3390/s21144771

Chen, R., Shiau, Y.-C., Chiu, Y.-P., & Wu, P.-Y. (2015). Applying QR codes to building facility management system. *ICIC Express Letters, Part B: Applications*, *6*(3), 749–756.

Chen, W., Chen, K., Cheng, J. C. P., Wang, Q., & Gan, V. J. L. (2018). BIM-based framework for automatic scheduling of facility maintenance work orders. *Automation in Construction*, *91*, 15–30. https://doi.org/10.1016/J.AUTCON.2018.03.007

Chen, W., Cheng, J. C. P., & Tan, Y. (2019). BIM- and IoT-based data-driven decision support system for predictive maintenance of building facilities. In *Innovative Production And Construction: Transforming Construction Through Emerging Technologies* (pp. 429–447). World Scientific Publishing Co. https://doi.org/10.1142/9789813272491_0025

Chen, Y.-J., Lai, Y.-S., & Lin, Y.-H. (2020). BIM-based augmented reality inspection and maintenance of fire safety equipment. *Automation in Construction*, *110*, 103041. https://doi.org/10.1016/j.autcon.2019.103041

Cheng, B., & Titterington, D. M. (1994). Neural networks: a review from a statistical perspective. *Statistical Science*, *9*(1), 2–30. https://doi.org/10.1214/ss/1177010638

Cheng, G., Li, Y., Gao, Z., & Liu, X. (2017). Cloud data governance maturity model. *2017 8th IEEE International Conference on Software Engineering and Service Science (ICSESS)*, 517–520. https://doi.org/10.1109/ICSESS.2017.8342968

Cheng, J. C. P., Chen, W., Chen, K., & Wang, Q. (2020). Data-driven predictive maintenance planning framework for MEP components based on BIM and IoT using machine learning algorithms. *Automation in Construction*, *112*. https://doi.org/10.1016/j.autcon.2020.103087

Chou, C. C., Chiang, C. T., Lin, H. P., Chu, C. P., & Lin, C. Y. (2016). Expansion of Building Information Models using spatial databases and serious game platforms for facility operations and maintenance management. *Journal of the Chinese Institute of Civil and Hydraulic Engineering*, *28*(4), 255–266. https://doi.org/10.6652/JoCICHE/2016-02804-03

Chung, S. W., Kwon, S. W., Moon, D. Y., & Ko, T. K. (2018). Smart facility management systems utilizing open BIM and augmented/virtual reality. *34th International Symposium on Automation and Robotics in Construction*, 846–853. https://doi.org/10.22260/isarc2018/0118

Clemmensen, L. H., & Kjærsgaard, R. D. (2022). Data representativity for machine learning and ai systems. *ArXiv Preprint ArXiv:2203.04706*, 1–28. https://doi.org/10.48550/arXiv.2203.04706

Cong, Z., Yin, H., Manzoor, F., Cahill, B., & Menzel, K. (2010). Integration of radio frequency identification and building information modelling for decentralised information management. In T. W. (Ed.), *Proceedings of 13th International Conference on Computing in Civil and Building Engineering*. Nottingham. https://silo.tips/download/integration-of-radio-frequency-identification-and-building-information-modelling

Corneli, A., Naticchia, B., Carbonari, A., & Bosché, F. (2019). Augmented reality and deep learning towards the management of secondary building assets. In A.-H. M. (Ed.), *36th International Symposium on Automation and Robotics in Construction* (pp. 332–339). International Association for Automation and Robotics in Construction I.A.A.R.C). https://doi.org/10.22260/ISARC2019/0045

Dahanayake, K. C., & Sumanarathna, N. (2022). IoT-BIM-based digital transformation in facilities management: a conceptual model. *Journal of Facilities Management*, *20*(3), 437–451. https://doi.org/10.1108/JFM-10-2020-0076

Daqi, G., & Yan, J. (2005). Classification methodologies of multilayer perceptrons with sigmoid activation functions. *Pattern Recognition*, *38*(10), 1469–1482. https://doi.org/10.1016/j.patcog.2005.03.024

Delgoshaei, P., Heidarinejad, M., & Austin, M. A. (2022). A semantic approach for building system operations: knowledge representation and reasoning. In *Sustainability* (Vol. 14, Issue 10). https://doi.org/10.3390/su14105810

Dellermann, D., Calma, A., Lipusch, N., Weber, T., Weigel, S., & Ebel, P. (2019). The future of human-AI collaboration: a taxonomy of design knowledge for hybrid intelligence systems. *Hawaii International Conference on System Sciences*. https://doi.org/10.24251/HICSS.2019.034

Dellermann, D., Ebel, P., Söllner, M., & Leimeister, J. M. (2019). Hybrid intelligence. *Business & Information Systems Engineering*, *61*(5), 637–643. https://doi.org/10.1007/s12599-019-00595-2

Diao, P.-H., & Shih, N.-J. (2019). BIM-based AR maintenance system (BARMS) as an intelligent instruction platform for complex plumbing facilities. *Applied Sciences (Switzerland)*, *9*(8). https://doi.org/10.3390/app9081592

Dias, P. D. R., & Ergan, S. (2020). Owner requirements in as-built BIM deliverables and a system architecture for FM-specific BIM representation. *Canadian Journal of Civil Engineering*, *47*(2), 215–227. https://doi.org/10.1139/cjce-2018-0703

Dias, P., & Ergan, S. (2016). *The Need for Representing Facility Information with Customized LOD for Specific FM Tasks* (P.-R. J.L., L. del P. C., G.-Q. A., M.-F. F., & M.-B. O.I. (eds.); pp. 2563–2572). American Society of Civil Engineers (ASCE). https://doi.org/10.1061/9780784479827.255

Dibowski, H., Holub, O., & Rojícek, J. (2016). Knowledge-based fault propagation in building automation systems. *2016 International Conference on Systems Informatics, Modelling and Simulation (SIMS)*, 124–132. https://doi.org/10.1109/SIMS.2016.22

Dong, B., O'Neill, Z., & Li, Z. (2014). A BIM-enabled information infrastructure for building energy Fault Detection and Diagnostics. *Automation in Construction*, *44*, 197–211. https://doi.org/10.1016/j.autcon.2014.04.007

East, B. (2011). *Construction Operations Building Information Exchange (COBie)*.

Eastman, C., Teicholz, P., Sacks, R., & Liston, K. (2011). *BIM handbook: a guide to building information modeling for owners, managers, designers, engineers and contractors* (Vol. 2). John Wiley and Sons, Inc. https://doi.org/10.1002/9780470261309

Edirisinghe, R., London, K. A., Kalutara, P., & Aranda-Mena, G. (2017). Building information modelling for facility management: are we there yet? *Engineering, Construction and Architectural Management*, *24*(6), 1119–1154. https://doi.org/10.1108/ECAM-06-2016-0139

Edirisinghe, R., Setunge, S., & Zhang, G. (2013). Application of gamma process for building deterioration prediction. *Journal of Performance of Constructed Facilities*, *27*(6), 763–773. https://doi.org/10.1061/(ASCE)CF.1943-5509.0000358

El Ammari, K., & Hammad, A. (2019). Remote interactive collaboration in facilities management using BIM-based mixed reality. *Automation in Construction, 107*. https://doi.org/10.1016/j.autcon.2019.102940

Ensafi, M., Harode, A., & Thabet, W. (2022). Developing systems-centric as-built BIMs to support facility emergency management: a case study approach. *Automation in Construction*, *133*. https://doi.org/10.1016/j.autcon.2021.104003

Ensafi, M., & Thabet, W. (2021). Challenges and gaps in facility maintenance practices. In T. Leathem, A. Perrenoud, & W. Collins (Eds.), *57th Annual Associated Schools of Construction International Conference* (Vol. 2, Issue July, pp. 237–227). https://doi.org/10.29007/1h2j

Ensafi, M., Thabet, W., Afsari, K., & Yang, E. (2023). Challenges and gaps with user-led decision-making for prioritizing maintenance work orders. *Journal of Building Engineering*, *66*, 105840. https://doi.org/https://doi.org/10.1016/j.jobe.2023.105840

Farghaly, K., Abanda, H., Vidalakis, C., & Wood, G. (2017). *BIM for asset management: A taxonomy of non-geometric BIM data for asset management* (T. W., K. C., & N. J. (eds.); pp. 96–105). European Group for Intelligent Computing in Engineering (EG-ICE). https://www.scopus.com/inward/record.uri?eid=2-s2.0-85026801747&partnerID=40&md5=c221361152828a8892fac974fb9c119d

Feng, C.-W., Chen, Y.-J., & Huang, S.-H. (2019). Applying AR and BIM in the spatial analysis of maintenance work of eletromechanical facilities. *Journal of the Chinese Institute of Civil and Hydraulic Engineering*, *31*(2), 163–172. https://doi.org/10.6652/JoCICHE.201904_31(2).0004

Fernald, H., Hong, S., Bucking, S., & O’Brien, W. (2018). BIM to BEM translation workflows and their challenges: a case study using a detailed BIM model. *Proceedings of ESim 2018, the 10th Conference of IBPSA-Canada*, 482–491.

https://publications.ibpsa.org/proceedings/esim/2018/papers/esim2018_2-3-A-3.pdf

Finco, F., Oliveira, A., Sousa, N., Pinto, C., Granja, J., & Azenha, M. (2023). Development of a BIM model for facility management with virtual/augmented reality interaction. In A. Gomes Correia, M. Azenha, P. J. S. Cruz, P. Novais, & P. Pereira (Eds.), *Trends on Construction in the Digital Era* (pp. 215–232). Springer International Publishing.

Firas, S., & Yelda, T. (2017). *A semi-automated approach for detecting building spaces with deteriorating performance using IFC-BIM and energy simulations*. *2017-May*, 45–53. https://www.scopus.com/inward/record.uri?eid=2-s2.0-85053618593&partnerID=40&md5=45a52e01182e59f951f7642c88544379

Florez, L., & Afsari, K. (2018). *Integrating facility management information into building information modelling using COBie: Current status and future directions*. https://doi.org/10.22260/isarc2018/0116

Follini, C., Magnago, V., Freitag, K., Terzer, M., Marcher, C., Riedl, M., Giusti, A., & Matt, D. T. (2021). BIM-integrated collaborative robotics for application in building construction and Maintenance. In *Robotics* (Vol. 10, Issue 1, p. 2). https://doi.org/10.3390/robotics10010002

Fraser, K. (2014). Facilities management: the strategic selection of a maintenance system. *Journal of Facilities Management*, *12*(1), 18–37. https://doi.org/10.1108/JFM-02-2013-0010

Gallaher, M. P., O'Conor, A. C., Dettbarn, J. L., & Gilday, L. T. (2004). Cost analysis of inadequate interoperability in the U.S. capital facilities industry. In *Nist*. https://doi.org/10.6028/NIST.GCR.04-867

Gao, X., & Pishdad-Bozorgi, P. (2018). Past, present, and future of BIM-enabled facilities operation and maintenance. *Construction Research Congress 2018*, *2018-April*, 51–61. https://doi.org/10.1061/9780784481295.006

Gao, X., & Pishdad-Bozorgi, P. (2019). BIM-enabled facilities operation and maintenance: A review. *Advanced Engineering Informatics*, *39*, 227–247. https://doi.org/10.1016/j.aei.2019.01.005

Ghosh, A., Chasey, A. D., & Mergenschroer, M. (2015). Building information modeling for facilities management: current practices and future prospects. In *Building Information Modeling* (pp. 223–253). https://doi.org/doi:10.1061/9780784413982.ch09

Gökalp, M. O., Gökalp, E., Kayabay, K., Koçyiğit, A., & Eren, P. E. (2022). The development of the data science capability maturity model: a survey-based research. *Online Information Review*, *46*(3), 547–567. https://doi.org/10.1108/OIR-10-2020-0469

Golabchi, A., Akula, M., & Kamat, V. (2016). Automated building information modeling for fault detection and diagnostics in commercial HVAC systems. *Facilities*, *34*(3–4), 233–246. https://doi.org/10.1108/F-06-2014-0050

Golabchi, A., Akula, M., & Kamat, V. R. (2013). Leveraging BIM for automated fault detection in operational buildings. In C. Haas (Ed.), *30th International Symposium on Automation and Robotics in Construction and Mining (ISARC 2013): Building the Future in Automation and Robotics* (pp. 187–197). International Association for Automation and Robotics in Construction (IAARC). https://doi.org/10.22260/ISARC2013/0020

Granderson, J., & Lin, G. (2019). *Inventory of data sets for AFDD evaluation*.

Gu, B., Ergan, S., & Akinci, B. (2014). Generating as-is building information models for facility management by leveraging heterogeneous existing information sources: A case study. *Construction Research Congress 2014: Construction in a Global Network*, 1911–1920. https://doi.org/10.1061/9780784413517.0195

Guerrero, J. I., Miró-Amarante, G., & Martín, A. (2022). Decision support system in health care building design based on case-based reasoning and reinforcement learning. *Expert Systems with Applications*, *187*, 116037. https://doi.org/10.1016/j.eswa.2021.116037

Guo, Y., Li, G., Chen, H., Wang, J., Guo, M., Sun, S., & Hu, W. (2017). Optimized neural network-based fault diagnosis strategy for VRF system in heating mode using data mining. *Applied Thermal Engineering*, *125*, 1402–1413. https://doi.org/10.1016/j.applthermaleng.2017.07.065

Halmetoja, E. (2019). The conditions data model supporting building information models in facility management. *Facilities*, *37*(7–8), 484–501. https://doi.org/10.1108/F-11-2017-0112

Halmetoja, E. (2022). The Role of Digital Twins and their application for the built environment. In *Structural Integrity* (Vol. 20, pp. 415–442). Springer Science and Business Media Deutschland GmbH. https://doi.org/10.1007/978-3-030-82430-3_18

Halmetoja, E., & Lepkova, N. (2022). Utilising Building Information Models in Facility Maintenance and Operations TT - Utilising Building Information Models in Facility Maintenance and Operations. *Teknik Dergi*, *33*(5), 12337–12351. https://doi.org/10.18400/tekderg.748397

Heaton, J., Parlikad, A. K., & Schooling, J. (2019). Design and development of BIM models to support operations and maintenance. *Computers in Industry*, *111*, 172–186. https://doi.org/10.1016/j.compind.2019.08.001

Hecht-Nielsen. (1989). Theory of the backpropagation neural network. *International 1989 Joint Conference on Neural Networks*, 593–605 vol.1. https://doi.org/10.1109/IJCNN.1989.118638

Hicks, B. J. (2007). Lean information management: Understanding and eliminating waste. *International Journal of Information Management*, *27*(4), 233–249. https://doi.org/10.1016/j.ijinfomgt.2006.12.001

Himmelblau, D. M. (1978). Fault detection ond diagnosis in chemical and petrochemicol processes. In *Scientific Publishing Co.* Scientific Publishing Co. https://doi.org/https://doi.org/10.1002/aic.690250527

Hosamo, H. H., Imran, A., Cardenas-Cartagena, J., Svennevig, P. R., Svidt, K., & Nielsen, H. K. (2022). A Review of the Digital Twin technology in the AEC-FM industry. *Advances in Civil Engineering*, *2022*, 2185170. https://doi.org/10.1155/2022/2185170

Hosamo, H. H., Nielsen, H. K., Alnmr, A. N., Svennevig, P. R., & Svidt, K. (2022). A review of the Digital Twin technology for fault detection in buildings. In *Frontiers in Built Environment* (Vol. 8, p. 1013196). https://www.frontiersin.org/articles/10.3389/fbuil.2022.1013196

Hosamo, H. H., Nielsen, H. K., Kraniotis, D., Svennevig, P. R., & Svidt, K. (2023). Digital Twin framework for automated fault source detection and prediction for comfort performance evaluation of existing non-residential Norwegian buildings. *Energy and Buildings*, *281*, 112732. https://doi.org/10.1016/j.enbuild.2022.112732

Hosamo, H. H., Svennevig, P. R., Svidt, K., Han, D., & Nielsen, H. K. (2022). A Digital Twin predictive maintenance framework of air handling units based on automatic fault detection and diagnostics. *Energy and Buildings*, *261*. https://doi.org/10.1016/j.enbuild.2022.111988

Hou, Z., Lian, Z., Yao, Y., & Yuan, X. (2006). Data mining based sensor fault diagnosis and validation for building air conditioning system. *Energy Conversion and Management*, *47*(15), 2479–2490. https://doi.org/10.1016/j.enconman.2005.11.010

Hu, Z. Z., Tian, P. L., Li, S. W., & Zhang, J. P. (2018). BIM-based integrated delivery technologies for intelligent MEP management in the operation and maintenance phase. *Advances in Engineering Software*, *115*, 1–16. https://doi.org/10.1016/j.advengsoft.2017.08.007

International Facility Management Association. (2018). *What is facility management?* www.ifma.org/about/what-is-facility-management

International Organization for Standardization. (2018). *Industry Foundation Classes (IFC) for data sharing in the construction and facility management industries*.

Irizarry, J., Gheisari, M., Williams, G., & Roper, K. (2014). Ambient intelligence environments for accessing building information: A healthcare facility management scenario. *Facilities*, *32*(3), 120–138. https://doi.org/10.1108/F-05-2012-0034

Isermann, R. (2006). An introduction from fault detection to fault tolerance. In *Fault-diagnosis systems*. Springer, Berlin (Germany). https://doi.org/10.1007/3-540-30368-5

Janocha, K., & Czarnecki, W. M. (2017). On loss functions for deep neural networks in classification. *Schedae Informaticae*, *25*, 49–59. https://doi.org/10.4467/20838476SI.16.004.6185

Jiang, Z., Messner, J. I., & Dubler, C. R. (2017). Defining a taxonomy for virtual 3D city model use cases with a focus on facility asset management—A virtual campus case study. In *Computing in Civil Engineering 2017* (pp. 43–50). https://doi.org/10.1061/9780784480823.006

Jirgl, M., Bradac, Z., & Fiedler, P. (2018). Human-in-the-loop issue in context of the cyber-physical systems. *IFAC-PapersOnLine*, *51*(6), 225–230. https://doi.org/https://doi.org/10.1016/j.ifacol.2018.07.158

Kamal, Z., Taghaddos, H., & Karimi, H. (2021). BIM-based maintenance management system for healthcare facilities. *Journal of Performance of Constructed Facilities*, *35*(4), 4021036. https://doi.org/10.1061/(ASCE)CF.1943-5509.0001604

Kamar, E. (2016). Directions in hybrid intelligence: complementing AI systems with human intelligence. *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence*, 4070–4073. https://doi.org/10.5555/3061053.3061219

Kameli, M., Hosseinalipour, M., Majrouhi Sardroud, J., Ahmed, S. M., & Behruyan, M. (2021). Improving maintenance performance by developing an IFC BIM/RFID-based computer system. *Journal of Ambient Intelligence and Humanized Computing*, *12*(2), 3055–3074. https://doi.org/10.1007/s12652-020-02464-3

Kane, M. B. (2018). Modeling human-in-the-loop behavior and interactions with HVAC systems. *2018 Annual American Control Conference (ACC)*, 4628–4633. https://doi.org/10.23919/ACC.2018.8431913

Kasprzak, C., Ramesh, A., & Dubler, C. (2013). *Developing standards to assess the quality of BIM criteria for facilities management*. 680–690. https://doi.org/10.1061/9780784412909.067

Katipamula, S., & Brambley, M. (2005). Review article: methods for fault detection, diagnostics, and prognostics for building systems—a review, Part I. *HVAC&R Research*, *11*(1), 3–25. https://doi.org/10.1080/10789669.2005.10391123

Kensek, K. (2015). BIM guidelines inform facilities management databases: a case study over time. *Buildings*, *5*(3), 899–916. https://doi.org/10.3390/buildings5030899

Khalek, I. A., Chalhoub, J. M., & Ayer, S. K. (2019). Augmented reality for identifying maintainability concerns during design. *Advances in Civil Engineering*, *2019*, 8547928. https://doi.org/10.1155/2019/8547928

Kim, K., Kim, H., Kim, W., Kim, C., Kim, J., & Yu, J. (2018). Integration of ifc objects and facility management work information using Semantic Web. *Automation in Construction, 87*, 173–187. https://doi.org/10.1016/j.autcon.2017.12.019

Kim, S., Poirier, E. A., & Staub-French, S. (2020). Information commissioning: bridging the gap between digital and physical built assets. *Journal of Facilities Management*, *18*(3), 231–245. https://doi.org/10.1108/JFM-04-2020-0024

Kim, W., & Katipamula, S. (2018). A review of fault detection and diagnostics methods for building systems. *Science and Technology for the Built Environment*, *24*(1), 3–21. https://doi.org/10.1080/23744731.2017.1318008

Koch, C., Neges, M., König, M., & Abramovici, M. (2012). BIM-based augmented reality for facility maintenance using natural markers. In B. A., R. Y., G. P., & de W. P. (Eds.), *Proceedings of the 2021 European Conference for Computing in Construction* (pp. 431–438). Universiteit Twente. https://doi.org/10.35490/EC3.2021.180

Konstantinou, I., Batzias, F., & Bountri, A. (2011). Integrating reliability, risk analysis and quality management in wastewater treatment facilities. *Proceedings of the 6th IASME/WSEAS International Conference on Energy \& Environment*, 111–116.

Korpela, J., Miettinen, R., Salmikivi, T., & Ihalainen, J. (2015). The challenges and potentials of utilizing building information modelling in facility management: the case of the Center for Properties and Facilities of the University of Helsinki. *Construction Management and Economics*, *33*(1), 3–17. https://doi.org/10.1080/01446193.2015.1016540

Lavy, S., & Jawadekar, S. (2014). A case study of Using BIM and COBie for facility management. *International Journal of Facility Management*, *5*(2), 13–27. http://ijfm.net/index.php/ijfm/article/view/110/114

Lavy, S., & Saxena, N. (2015). *Work order processing times - Does bim make a difference...?* 565–572.

Lavy, S., Saxena, N., & Dixit, M. (2019). Effects of BIM and COBie Database Facility Management on Work Order Processing Times: Case Study. *Journal of Performance of Constructed Facilities*, *33*(6). https://doi.org/10.1061/(ASCE)CF.1943-5509.0001333

Liang, J., & Du, R. (2007). Model-based fault detection and diagnosis of HVAC systems using support vector machine method. *International Journal of Refrigeration*, *30*(6), 1104–1114. https://doi.org/10.1016/j.ijrefrig.2006.12.012

Lin, T. Y., Shih, H. Y., Huang, J. C., Hou, Y. L., Chiu, Y. Y., & Lin, Y. X. (2018). Study on the dormitory public facilities management using BIM. In M. T.-H. (Ed.), *2018 IEEE International Conference on Advanced Manufacturing* (pp. 191–193). Institute of Electrical and Electronics Engineers Inc. https://doi.org/10.1109/AMCON.2018.8614813

Lin, Y.-C., Chen, Y.-P., Huang, W.-T., & Hong, C.-C. (2016). Development of BIM execution plan for BIM model management during the pre-operation phase: a case study. *Buildings*, *6*(1). https://doi.org/10.3390/buildings6010008

Lin, Y. C., & Su, Y. C. (2014). Developing mobile- and BIM-based integrated visual facility maintenance management system. *The Scientific World Journal*, *2014*. https://doi.org/10.1155/2013/124249

Liu, F., Jallow, A. K., Anumba, C., & Wu, D. (2014). A framework for integrating change management with Building Information Modeling. In *Computing in Civil and Building Engineering (2014)* (pp. 439–446). https://doi.org/doi:10.1061/9780784413616.055

Liu, Q., & Gao, T. (2017). The information requirements for transportation industry’s facilities management based on BIM. *Open Construction and Building Technology Journal*, *11*, 136–141. https://doi.org/10.2174/1874836801711010136

Liu, R., & Issa, R. R. A. (2012). *3D visualization of sub-surface pipelines in connection with the building utilities: Integrating GIS and BIM for facility management*. 341–348. https://doi.org/10.1061/9780784412343.0043

Liu, R., & Issa, R. R. A. (2013a). BIM for facility management: design for maintainebility with BIM tools. *Proceedings of the 30th ISARC*, 321–328. https://www.scopus.com/inward/record.uri?eid=2-s2.0-84893548783&partnerID=40&md5=e42ff0197e04edb08727481cbe573521

Liu, R., & Issa, R. R. A. (2013b). *Issues in BIM for facility management from industry practitioners' perspectives*. 411–418. https://doi.org/10.1061/9780784413029.052

Liu, R., & Issa, R. R. A. (2014). Design for maintenance accessibility using BIM tools. *Facilities*, *32*(3), 153–159. https://doi.org/10.1108/F-09-2011-0078

Liu, R., & Issa, R. R. A. (2016). Survey: common knowledge in BIM for facility maintenance. *Journal of Performance of Constructed Facilities*, *30*(3), 4015033. https://doi.org/10.1061/(ASCE)CF.1943-5509.0000778

Lu, Q., Xie, X., Heaton, J., Parlikad, A. K., & Schooling, J. (2020). From BIM towards digital twin: strategy and future development for smart asset management. In T. Borangiu, D. Trentesaux, P. Leitão, A. Giret Boggino, & V. Botti (Eds.), *Service Oriented, Holonic and Multi-agent Manufacturing Systems for Industry of the Future* (pp. 392–404). Springer International Publishing.

Lucas, J., Bulbul, T., & Thabet, W. (2013a). An object-oriented model to support healthcare facility information management. *Automation in Construction*, *31*, 281–291. https://doi.org/https://doi.org/10.1016/j.autcon.2012.12.014

Lucas, J., Bulbul, T., & Thabet, W. (2013b). An object-oriented model to support healthcare facility information management. *Automation in Construction*, *31*, 281–291. https://doi.org/10.1016/j.autcon.2012.12.014

Lucas, J., Bulbul, T., Thabet, W., & Anumba, C. (2013a). A case study of using BIM and COBie for facility management. *Journal of Architectural Engineering*, *19*(2), 134–145. https://doi.org/10.1061/(ASCE)AE.1943-5568.0000111

Lucas, J., Bulbul, T., Thabet, W., & Anumba, C. (2013b). Case Analysis to Identify Information Links between Facility Management and Healthcare Delivery Information in a Hospital Setting. *Journal of Architectural Engineering*, *19*(2), 134–145. https://doi.org/10.1061/(ASCE)AE.1943-5568.0000111

Ma, Z., Ren, Y., Xiang, X., & Turk, Z. (2020). Data-driven decision-making for equipment maintenance. *Automation in Construction*, *112*. https://doi.org/10.1016/j.autcon.2020.103103

Malhotra, R. (2015). A systematic review of machine learning techniques for software fault prediction. *Applied Soft Computing*, *27*, 504–518. https://doi.org/10.1016/j.asoc.2014.11.023

Malla, V., Tummalapudi, M., & Delhi, V. S. K. (2024). Perceptions of built-environment professionals on using ISO 19650 standards for information management. *Journal of Legal Affairs and Dispute Resolution in Engineering and Construction*, *16*(1), 4523045. https://doi.org/10.1061/JLADAH.LADR-1076

Mamaghani, O. A., & Noorzai, E. (2023). A framework to implement augmented reality based on BIM to improve operation and maintenance of mechanical facilities of commercial complexes. *Facilities*, *41*(3/4), 229–247. https://doi.org/10.1108/F-04-2022-0064

Mayo, G., & Issa, R. R. A. (2016). Nongeometric Building Information Needs Assessment for Facilities Management. *Journal of Management in Engineering*, *32*(3). https://doi.org/10.1061/(ASCE)ME.1943-5479.0000414

McArthur, J. J., Shahbazi, N., Fok, R., Raghubar, C., Bortoluzzi, B., & An, A. (2018). Machine learning and BIM visualization for maintenance issue classification and enhanced data collection. *Advanced Engineering Informatics*, *38*, 101–112. https://doi.org/10.1016/j.aei.2018.06.007

Meadati, P., & Irizarry, J. (2015). *BIM and QR code for operation and maintenance* (O. W.J. & P. S. (eds.); Vols. 2015-Janua, Issue January, pp. 556–563). American Society of Civil Engineers (ASCE). https://doi.org/10.1061/9780784479247.069

Meadati, P., Irizarry, J., & Akhnoukh, A. (2011). *Building information modeling implementation - Current and desired status*. 512–519. https://doi.org/10.1061/41182(416)63

Meimand, M., & Jazizadeh, F. (2022). Human-in-the-loop model predictive operation for energy efficient HVAC systems. In *Construction Research Congress 2022* (pp. 178–187). https://doi.org/doi:10.1061/9780784483954.019

Mirnaghi, M. S., & Haghighat, F. (2020). Fault detection and diagnosis of large-scale HVAC systems in buildings using data-driven methods: A comprehensive review. *Energy and Buildings*, *229*, 110492. https://doi.org/10.1016/j.enbuild.2020.110492

Moon, K., Bergemann, P., Brown, D., Chen, A., Chu, J., Eisen, E. A., Fischer, G. M., Loyalka, P., Rho, S., & Cohen, J. (2022). Manufacturing productivity with worker turnover. *Management Science*, *69*(4), 1995–2015. https://doi.org/10.1287/mnsc.2022.4476

Moreno, J. V, Machete, R., Falcão, A. P., Gonçalves, A. B., & Bento, R. (2022). Dynamic Data Feeding into BIM for Facility Management: A Prototype Application to a University Building. *Buildings*, *12*(5). https://doi.org/10.3390/buildings12050645

Moretti, N., Blanco Cadena, J. D., Mannino, A., Poli, T., & Re Cecconi, F. (2021). Maintenance service optimization in smart buildings through ultrasonic sensors network. *Intelligent Buildings International*, *13*(1), 4–16. https://doi.org/10.1080/17508975.2020.1765723

Motamedi, A., Hammad, A., & Asen, Y. (2014). Knowledge-assisted BIM-based visual analytics for failure root cause detection in facilities management. *Automation in Construction*, *43*, 73–83. https://doi.org/10.1016/j.autcon.2014.03.012

Motamedi, A., Soltani, M. M., & Hammad, A. (2013a). Indoor localization of RFID-equipped movable assets using mobile reader based on reference tags clustering. *30th International Symposium on Automation and Robotics in Construction*, 626–634. https://doi.org/10.22260/isarc2013/0068

Motamedi, A., Soltani, M. M., & Hammad, A. (2013b). Localization of RFID-equipped assets during the operation phase of facilities. *Advanced Engineering Informatics*, *27*(4), 566–579. https://doi.org/10.1016/j.aei.2013.07.001

Motawa, I., & Almarshad, A. (2013). A knowledge-based BIM system for building maintenance. *Automation in Construction*, *29*, 173–182. https://doi.org/10.1016/j.autcon.2012.09.008

Natephra, W., & Motamedi, A. (2019). Live data visualization of IoT sensors using augmented reality (AR) and BIM. In A.-H. M. (Ed.), *Proceedings of the 36th The International Association for Automation and Robotics in Construction* (pp. 632–638). International Association for Automation and Robotics in Construction I.A.A.R.C). https://doi.org/10.22260/isarc2019/0084

Nojedehi, P., O'brien, W., & Gunay, H. B. (2021). Benchmarking and visualization of building portfolios by applying text analytics to maintenance work order logs. *Science and Technology for the Built Environment*, *27*(6), 756–775. https://doi.org/10.1080/23744731.2021.1913957

Nojedehi, P., O'Brien, W., & Gunay, H. B. (2022). A methodology to integrate maintenance management systems and BIM to improve building management. *Science and Technology for the Built Environment*. https://doi.org/10.1080/23744731.2022.2052668

Oyediran, H., Ghimire, P., Peavy, M., Kim, K., & Barutha, P. (2021). Robotics applicability for routine operator tasks in power plant facilities. *38th International Symposium on Automation and Robotics in Construction*, *2021-Novem*, 677–682. https://doi.org/10.22260/ISARC2021/0091

Pan, N.-H., & Chen, K.-Y. (2020). Facility Maintenance Traceability Information Coding in BIM-Based Facility Repair Platform. *Advances in Civil Engineering*, *2020*. https://doi.org/10.1155/2020/3426563

Pan, X., Mateen Khan, A., Eldin, S. M., Aslam, F., Kashif Ur Rehman, S., & Jameel, M. (2024). BIM adoption in sustainability, energy modelling and implementing using ISO 19650: A review. *Ain Shams Engineering Journal*, *15*(1), 102252. https://doi.org/https://doi.org/10.1016/j.asej.2023.102252

Patacas, J., Dawood, N., Greenwood, D., & Kassem, M. (2016). Supporting building owners and facility managers in the validation and visualisation of asset information models (aim) through open standards and open technologies. *Journal of Information Technology in Construction*, *21*, 434–455. https://www.scopus.com/inward/record.uri?eid=2-s2.0-85001043352&partnerID=40&md5=a4c362c19fe5abdd5d0f4ecad5826ac0

Patil, A., Soni, G., Prakash, A., & Karwasra, K. (2022). Maintenance strategy selection: a comprehensive review of current paradigms and solution approaches. *International Journal of Quality & Reliability Management*, *39*(3), 675–703. https://doi.org/10.1108/IJQRM-04-2021-0105

Pérez-Sánchez, B., Fontenla-Romero, O., & Guijarro-Berdiñas, B. (2018). A review of adaptive online learning for artificial neural networks. *Artificial Intelligence Review*, *49*(2), 281–299. https://doi.org/10.1007/s10462-016-9526-2

Petersen, L., Minkkinen, P., & Esbensen, K. H. (2005). Representative sampling for reliable data analysis: theory of sampling. *Chemometrics and Intelligent Laboratory Systems*, *77*(1), 261–277. https://doi.org/10.1016/j.chemolab.2004.09.013

Pinti, L., Bonelli, S., Brizzolari, A., Mirarchi, C., Dejaco, M. C., & Kiviniemi, A. (2018). Integrated information management for the fm: Building information modelling and database integration for the italian public administration. In K. J. & S. R. (Eds.), *12th European Conference on Product & Process Modelling* (pp. 21–28). CRC Press/Balkema. https://doi.org/10.1201/9780429506215-3

Ravi, S., Md, T. I., Bashima, I., Zeyu, W., Tamim, S., Omprakash, G., & Shahriar, N. (2017). Preventive maintenance of centralized HVAC Systems: use of acoustic sensors, feature extraction, and unsupervised learning. In *Proceedings of Building Simulation 2017: 15th Conference of IBPSA* (Vol. 15, pp. 2518–2524). IBPSA. https://doi.org/10.26868/25222708.2017.715

Renjith, V. R., Kalathil, M. J., Kumar, P. H., & Madhavan, D. (2018). Fuzzy FMECA (failure mode effect and criticality analysis) of LNG storage facility. *Journal of Loss Prevention in the Process Industries*, *56*, 537–547. https://doi.org/10.1016/j.jlp.2018.01.002

Rosemann, M., & De Bruin, T. (2005). Towards a business process management maturity model. *ECIS 2005 Proceedings of the Thirteenth European Conference on Information Systems*, 1–12. https://eprints.qut.edu.au/25194/

Sadeghi, M., Mehany, M., & Strong, K. (2018). *Integrating building information models and building operation information exchange systems in a decision support framework for facilities management* (W. C., B. C., H. C., L. Y., & H. R. (eds.); Vols. 2018-April, pp. 770–779). American Society of Civil Engineers (ASCE). https://doi.org/10.1061/9780784481295.077

Saka, A. B., Oyedele, L. O., Akanbi, L. A., Ganiyu, S. A., Chan, D. W. M., & Bello, S. A. (2023). Conversational artificial intelligence in the AEC industry: a review of present status, challenges and opportunities. *Advanced Engineering Informatics*, *55*, 101869. https://doi.org/10.1016/j.aei.2022.101869

Sattenini, A., Azhar, S., & Thuston, J. (2011). Preparing a building information model for facility maintenance and management. *Proceedings of the 28th ISARC*, 150–155. https://doi.org/10.22260/isarc2011/0024

Seghezzi, E., Di Giuda, G. M., Schievano, M., & Paleari, F. (2020). Bim-enabled facility management optimization based on post-occupancy evaluations and building monitoring: framework and first results. In A. H., Y. S., & S. A. (Eds.), *Proceedings of International Structural Engineering and Construction* (Vol. 7, Issue 2, p. FAM-02-1-FAM-02-6). ISEC Press. https://doi.org/10.14455/ISEC.2020.7(2).FAM-02

Shalabi, F., & Turkan, Y. (2017). IFC BIM-based facility management approach to optimize data collection for corrective maintenance. *Journal of Performance of Constructed Facilities*, *31*(1). https://doi.org/10.1061/(ASCE)CF.1943-5509.0000941

Shalabi, F., & Turkan, Y. (2020). Bim-energy simulation approach for detecting building spaces with faults and problematic behavior. *Journal of Information Technology in Construction*, *25*, 342–360. https://doi.org/10.36680/J.ITCON.2020.020

Shi, Z., & O'Brien, W. (2019). Development and implementation of automated fault detection and diagnostics for building systems: A review. *Automation in Construction*, *104*, 215–229. https://doi.org/10.1016/J.AUTCON.2019.04.002

Shu-Hsien Liao. (2005). Expert system methodologies and applications - a decade review from 1995 to 2004. *Expert Systems with Applications*, *28*(1), 93–103. https://doi.org/10.1016/j.eswa.2004.08.003

Singh, V., Mathur, J., & Bhatia, A. (2022). A comprehensive review: Fault detection, diagnostics, prognostics, and fault modeling in HVAC systems. *International Journal of Refrigeration*, *144*, 283–295. https://doi.org/10.1016/j.ijrefrig.2022.08.017

Slongo, C., Malacarne, G., & Matt, D. T. (2022). *The ifc file format as a means of integrating bim and gis: The case of the management and maintenance of underground networks*. *5*(4), 301–309. https://doi.org/10.5194/isprs-Annals-V-4-2022-301-2022

Smillie, R. J., Nugent, W. A., Sander, S. I., & Johnson, D. M. (1988). A comparative assessment of paper-based and computer-based maintenance information delivery systems. *Navy Personnel Research and Development Center, San Diego, CA.*

Song, J. W., Lee, K. H., Jeong, M. K., Lee, S. J., & Kwon, S. W. (2020). *MR-based equipment remote control and 3D digital working guidance for field-oriented maintenance*. 661–668. https://doi.org/10.22260/ISARC2020/0093

Standardisation, E. C. for. (2010). *EN 13306:2010 maintenance — maintenance terminology*.

Sun, C., Wang, S., & Li, L. (2021). *BIM-RFID technology for facility management in power systems*. *781*(4). https://doi.org/10.1088/1755-1315/781/4/042017

Szandała, T. (2021). Review and comparison of commonly used activation functions for deep neural networks. In A. K. Bhoi, P. K. Mallick, C.-M. Liu, & V. E. Balas (Eds.), *Bio-inspired Neurocomputing* (pp. 203–224). Springer Singapore. https://doi.org/10.1007/978-981-15-5495-7_11

Tehrani, B. M., Wang, J., & Wang, C. (2019). Review of human-in-the-loop cyber-physical systems (hilcps): the current status from human perspective. In *Computing in Civil Engineering 2019* (pp. 470–478). https://doi.org/doi:10.1061/9780784482438.060

Teicholz, P. (2018). BIM for facility managers. In *BIM for Facility Managers*. wiley. https://doi.org/10.1002/9781119572633

Thabet, W., & Lucas, J. (2017a). Asset data handover for a large educational institution: case-study approach. *Journal of Construction Engineering and Management*, *143*(11). https://doi.org/10.1061/(ASCE)CO.1943-7862.0001389

Thabet, W., & Lucas, J. D. (2017b). A 6-step systematic process for model-based facility data delivery. *Journal of Information Technology in Construction*, *22*, 104–131. https://www.scopus.com/inward/record.uri?eid=2-s2.0-85026328344&partnerID=40&md5=10e1ed7234ac4e14f6a09f27d6df8780

Thinh, T. Van, Duong, P. C., Nasahara, K. N., & Tadono, T. (2019). How does land use/land cover map's accuracy depend on number of classification classes? *SOLA*, *15*, 28–31. https://doi.org/10.2151/sola.2019-006

Thumati, B. T., Feinstein, M. A., Fonda, J. W., Turnbull, A., Weaver, F. J., Calkins, M. E., & Jagannathan, S. (2011). An online model-based fault diagnosis scheme for HVAC systems. *2011 IEEE International Conference on Control Applications (CCA)*, 70–75. https://doi.org/10.1109/CCA.2011.6044486

Vach, K., Holubec, P., & Dlesk, A. (2018). *New trends in GIS and BIM for facility management in the Czech Republic* (S. S., P. H., & K. A.S. (eds.); Vol. 42, Issue 5, pp. 135–138). International Society for Photogrammetry and Remote Sensing. https://doi.org/10.5194/isprs-archives-XLII-5-135-2018

Valdepeñas, P., Pérez, M. D. E., Henche, C., Rodríguez-Escribano, R., Fernández, G., & López-Gutiérrez, J.-S. (2020). Application of the BIM method in the management of the maintenance in port infrastructures. *Journal of Marine Science and Engineering*, *8*(12), 1–22. https://doi.org/10.3390/jmse8120981

Valinejadshoubi, M., Moselhi, O., & Bagchi, A. (2022). Integrating BIM into sensor-based facilities management operations. *Journal of Facilities Management*, *20*(3), 385–400. https://doi.org/10.1108/JFM-08-2020-0055

Venkatasubramanian, V., Rengaswamy, R., Kavuri, S. N., & Yin, K. (2003). A review of process fault detection and diagnosis: Part III: process history based methods. *Computers & Chemical Engineering*, *27*(3), 327–346. https://doi.org/10.1016/S0098-1354(02)00162-X

Villa, V., Bruno, G., Aliev, K., Piantanida, P., Corneli, A., & Antonelli, D. (2022). Machine learning framework for the sustainable maintenance of building facilities. *Sustainability (Switzerland)*, *14*(2). https://doi.org/10.3390/su14020681

Villa, V., Naticchia, B., Bruno, G., Aliev, K., Piantanida, P., & Antonelli, D. (2021). Iot open-source architecture for the maintenance of building facilities. *Applied Sciences (Switzerland)*, *11*(12). https://doi.org/10.3390/app11125374

Wan Siti Hajar, W. N., Syahrul Nizam, K., & Nurshuhada, Z. (2022). *Systematic Review: Information for FM-Enabled BIM* (Y. L., N. U., A. O.G., & A. M. (eds.); Vol. 161, pp. 133–150). Springer Science and Business Media Deutschland GmbH. https://doi.org/10.1007/978-981-16-2329-5_16

Wang, M., Altaf, M. S., Al-Hussein, M., & Ma, Y. (2020). Framework for an IoT-based shop floor material management system for panelized homebuilding. *International Journal of Construction Management*, *20*(2), 130–145. https://doi.org/10.1080/15623599.2018.1484554

Wang, S., & Xiao, F. (2004). AHU sensor fault diagnosis using principal component analysis method. *Energy and Buildings*, *36*(2), 147–160. https://doi.org/10.1016/j.enbuild.2003.10.002

Wang, T. K., & Piao, Y. (2019). Development of BIM-AR-based facility risk assessment and maintenance system. *Journal of Performance of Constructed Facilities*, *33*(6). https://doi.org/10.1061/(ASCE)CF.1943-5509.0001339

Wang, X., & Xie, M. (2022). Integration of 3DGIS and BIM and its application in visual detection of concealed facilities. *Geo-Spatial Information Science*. https://doi.org/10.1080/10095020.2022.2054732

Wang, Y., Wang, X., Wang, J., Yung, P., & Jun, G. (2013). Engagement of facilities management in design stage through BIM: framework and a case study. *Advances in Civil Engineering*, *2013*. https://doi.org/10.1155/2013/189105

Wang, Z., Bulbul, T., & Lucas, J. (2015). *A case study of BIM-based model adaptation for healthcare facility management - Information needs analysis* (O. W.J. & P. S. (eds.); Vols. 2015-Janua, Issue January, pp. 395–402). American Society of Civil Engineers (ASCE). https://doi.org/10.1061/9780784479247.049

Wanigarathna, N., Jones, K., Bell, A., & Kapogiannis, G. (2019). Building information modelling to support maintenance management of healthcare built assets. *Facilities*, *37*(7–8), 415–434. https://doi.org/10.1108/F-01-2018-0012

Wijekoon, C., Manewa, A., & Ross, A. D. (2020). Enhancing the value of facilities information management (FIM) through BIM integration. *Engineering, Construction and Architectural Management*, *27*(4), 809–824. https://doi.org/10.1108/ECAM-02-2016-0041

Wireman, T. (2009). *Successfully utilizing CMMS/EAM systems*. Industrial Press. http://www.books24x7.com/marc.asp?bookid=26011

Xie, X., Lu, Q., Rodenas-Herraiz, D., Parlikad, A. K., & Schooling, J. M. (2020). Visualised inspection system for monitoring environmental anomalies during daily operation and maintenance. *Engineering, Construction and Architectural Management*, *27*(8), 1835–1852. https://doi.org/10.1108/ECAM-11-2019-0640

Yan, K., Shen, W., Mulumba, T., & Afshari, A. (2014). ARX model based fault detection and diagnosis for chillers using support vector machines. *Energy and Buildings*, *81*, 287–295. https://doi.org/10.1016/j.enbuild.2014.05.049

Yan, R., Ma, Z., Kokogiannakis, G., & Zhao, Y. (2016). A sensor fault detection strategy for air handling units using cluster analysis. *Automation in Construction*, *70*, 77–88. https://doi.org/10.1016/j.autcon.2016.06.005

Yan, R., Ma, Z., Zhao, Y., & Kokogiannakis, G. (2016). A decision tree based data-driven diagnostic strategy for air handling units. *Energy and Buildings*, *133*, 37–45. https://doi.org/10.1016/j.enbuild.2016.09.039

Yang, X., & Ergan, S. (2016a). Leveraging BIM to provide automated support for efficient troubleshooting of HVAC-related problems. *Journal of Computing in Civil Engineering*, *30*(2), 04015023. https://doi.org/10.1061/(ASCE)CP.1943-5487.0000492

Yang, X., & Ergan, S. (2016b). Design and evaluation of an integrated visualization platform to support corrective maintenance of HVAC problem–related work orders. *Journal of Computing in Civil Engineering*, *30*(3), 04015041. https://doi.org/10.1061/(ASCE)CP.1943-5487.0000510

Yang, X., & Ergan, S. (2017). BIM for FM: information requirements to support HVAC-related corrective maintenance. *Journal of Architectural Engineering*, *23*(4). https://doi.org/10.1061/(ASCE)AE.1943-5568.0000272

Zaim, S., Turkyılmaz, A., Acar, M. F., Al-Turki, U., & Demirel, O. F. (2012). Maintenance strategy selection using AHP and ANP algorithms: a case study. *Journal of Quality in Maintenance Engineering*, *18*(1), 16–29. https://doi.org/10.1108/13552511211226166

Zeiler, W., & Labeodan, T. (2019). A human-in-the-loop approach for energy flexibility system integration to support infrastructures. In T. Ahram, W. Karwowski, & R. Taiar (Eds.), *International Conference on Human Systems Engineering and Design: Future Trends and Applications* (pp. 1027–1032). Springer International Publishing. https://doi.org/10.1007/978-3-030-02053-8_156

Zeiler, W., Vissers, D., Maaijen, R., & Boxem, G. (2014). Occupants' behavioural impact on energy consumption: 'human-in-the-loop' comfort process control. *Architectural Engineering and Design Management*, *10*(1–2), 108–130. https://doi.org/10.1080/17452007.2013.837252

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

Zhang, T., Doan, D. T., & Kang, J. (2023). Application of building information modeling-blockchain integration in the architecture, engineering, and construction / facilities management industry: a review. *Journal of Building Engineering*, *77*, 107551. https://doi.org/https://doi.org/10.1016/j.jobe.2023.107551

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*, 85–101. https://doi.org/10.1016/J.RSER.2019.04.021

Zhao, Y., Wang, S., Xiao, F., & Ma, Z. (2013). A simplified physical model-based fault detection and diagnosis strategy and its customized tool for centrifugal chillers. *HVAC&R Research*, *19*(3), 283–294. https://doi.org/10.1080/10789669.2013.765299

Zhou, Q., Wang, S., & Xiao, F. (2009). A novel strategy for the fault detection and diagnosis of centrifugal chiller systems. *HVAC&R Research*, *15*(1), 57–75. https://doi.org/10.1080/10789669.2009.10390825

Zimmermann, G., Lu, Y., & Lo, G. (2011). A simulation based fault diagnosis strategy using extended heat flow models (HFM). *Proceedings Building Simulation 2011*, 405–412. https://publications.ibpsa.org/proceedings/bs/2011/papers/bs2011_1231.pdf

Zimmermann, G., Lu, Y., & Lo, G. (2012). Automatic HVAC fault detection and diagnosis system generation based on heat flow models. *HVAC&R Research*, *18*(1–2), 112–125. https://doi.org/10.1080/10789669.2011.610427

APPENDICES

A. Statistical information of reviewed articles in Chapter 2

Tablo A.1 Statistics of categorization for the reviewed article

Category	Content	# of studies
FMM	Information needs & requirements, and quality of information	55
	Accessibility	3
	Benefits, values, and challenges: Case studies & handover	46
	Strategies for BIM-FMM	2
BIM-FMM Studies	BIM-FMM systems	33
	Applications in different areas for BIM-FM/M	46
	Interoperability & open BIM	70
	Integrated information technologies	97
	Legal issues	1
Fault management	Work order management	10
	Fault detection, diagnosis and reasoning	25
	Maintenance actions & practice and safety	1
	Data analysis and management for O&M	19
	Conditional assessments	13
	Root-cause analysis	5
BIM-FM Studies	BIM-FM & Digital Twin	90
	Location-based solutions	2
	Asset management	20
	Space management	1
Others	Review studies	33
	Others	16
TOTAL		423

B. Asset information parameters of CMMS tools in Chapter 2

Tablo B.2 Asset information parameters of CMMS tools

Tools	Parameters
Maximo	ID, description, status, model, serial number, type, location, current health condition, criticality, age in years, system, working calendar, working shift, vendor, manufacturer, installation date, expected life, purchase price, replacement cost, priority, failure class, total downtime, risk, spare parts & subcomponents, spare parts quantities on hand, spare parts quantities issued, safety- hazard type, safety- hazard materials, safety- precautions to each hazard type, safety, specifications, relationship (source or target)
Ecodomus	ID, name, description, serial number, tag, barcode, area served, warranty duration, warranty start, warranty end, installation date, technical features
Limbo	ID, name, category, model, PM schedule, work orders/logs, parts, vendors, report
QRmaint	ID, name, model, serial number, type, location, system, photo, manufacturer, installation date, service provider, warranty expiration date, purchase price, cost, contact person, asset maintainer
Mobility Work	Name, description, photo, cost center (production), barcode, tags, linked equipment, manufacturer
MaintainX	Name, description, photo, location, barcode, vendor
UpKeep	Name, description, model, category, area, barcode, photo, depreciation details, responsible personnel, vendor, customer, warranty expiration date, service date, spare parts, attached files, location, parent asset (system), check-in/out

Tablo B.1 Asset information parameters of CMMS tools (cont'd)

Tools	Parameters
Fiix	Name, ID, description, make, model, serial number, barcode, category, location, system, notes
MainWinWin	ID, name, model, serial number, year, category, location, system, supplier name, installation date, useful life, investment, photo, maintenance contract, warranty date
AssetPanda	ID, description, type, location, status, serial number, responsible personnel, cost, check out, manufacturer, date purchased, useful life, replacement cost, next service date, subcomponents, depreciation details
FM Systems	ID, description, model, serial number, manufacturer, installation date, warranty in months, warranty expiration date, condition, tag number, relationship type, maintenance log, priority, cost

C. Cross-validation results of validation case studies in Chapter 3

The case studies were cross validated with decision trees:

Table C.3. Cross-validation results of validation case studies

Case #	Train and test sets	Training Accuracy (%)	Testing Accuracy (%)
Case 1	Train with sets 1 and 2, and test with set 3	98.93	97.70
	Train with sets 1 and 3, and test with set 2	99.01	98.10
	Train with sets 2 and 3, and test with set 1	99.07	97.81
Case 2	Train with sets 1 and 2, and test with set 3	98.01	95.57
	Train with sets 1 and 3, and test with set 2	97.95	95.36
	Train with sets 2 and 3, and test with set 1	98.04	95.71

D. Work order contents in CMMS tools reviewed in Chapter 4

Table D.4. Work order request parameters of CMMS Tools

Tools	Parameters
Limbe	Title, description, requester contact info, location, asset, photo, files attached
QRmaint	Title, description, photo, asset, location, assigned personnel, task calendar, tags, priority, work type (breakdown, failure, inspection, calibration, legalization), maintenance type (cm, pm, request)
UpKeep	Description, location, photo, priority, files attached
MainWinWin	Title, ID, date, urgency, status, employee, approved by, description
MicroMain	ID, description, status, asset, location, priority, category, schedule (issued date, start date, due date, completed date, closed date, estimated duration)
BIM-genie	Description, location, photo, files attached
Maximo	Request ID, requester, contact info, asset, location, priority, reported and affected date, target start and finish
FM Systems	Requested for, location, department, category, description, upload document, upload image, is this a health and safety issue?

Table D.5. Work order parameters of CMMS Tools

Tools	Parameters
Maximo	Work order ID, description, asset, location, priority, parent work orders, maintenance work type, failure category (HVAC, pipe, engineering), problem code (too hot/cold, lighting problem), work order status, work order status date/log-flow action, schedule (scheduled/target/actual start/end date), job plan, safety plan, contract, inspection form, inspection result, responsibility (reported by, supervised by, lead by, assigned workers) datasheet, specifications
Limbo	Name, description, downtime, instructions, assigned personnel, due date, maintenance type, spare parts, tools, customized tags
QRmaint	Title, description, photo, asset or location, assigned personnel, task calendar, tags, priority, work type (breakdown, failure, inspection, calibration, legalization), maintenance type (CM, PM, request)
Mobility Works	Asset, task description, planned start and end dates, assigned personnel, observer, tags, checklist
MaintainX	Title, description, procedure, photo, assigned to, due date and schedule, priority, location, asset, attached files, maintenance categories (damage, electric, inspection, mechanical, safety), vendor
UpKeep	Title, description, photo, due date, estimated duration, priority, maintenance categories (damage, electric, inspection, mechanical, safety), assigned personnel, location, asset, purchase order/spare part, assigned tasks, attached files, signature, created time and by

Table D.2. Work order parameters of CMMS Tools (cont'd)

Tools	Parameters
Fiix	Work order code, asset code, project name, work order description, work order task description, task completion date, task time estimated, task time spent, task assigned user, task completed by the user, work order suggested completion date, date created, work order completion notes, work order problem, work order root cause, work order solution, work order status, work order priority, work order maintenance type, charge department code, account code
MainWinWin	ID, description, asset, urgency, system, work type, contract, status, employee, approved by
Maintenance Connection	ID, requester, contact info, failure class, problem, cause, remedy, asset, location, category, priority, procedure, tasks, documents, special instructions, labor report, maintenance log, assigned personnel, contract, shift
FM Systems	Request ID, priority, is this a health and safety issue? urgency, location, status, category, activity, problem description, action taken, estimated and actual hours, estimated and actual cost, equipment id, cause of incident, comments
A BIM-based work order tool	ID, WO site, issued date, status, maintenance organization, priority, criticality, work type, action, requester contract, schedule (planned/actual start and end date, time spent), event, discovery-error, symptom, contract, fault description, work description, error cause, inspection note, work done, assigned personnel (prepared by, work leader, executed by, contract no), asset, location, PM repeat details.

E. Sample examples of model-base work order generation in the Dynamo interface described in Chapter 4

E.1. Work order request:

Step 1: Select the space.

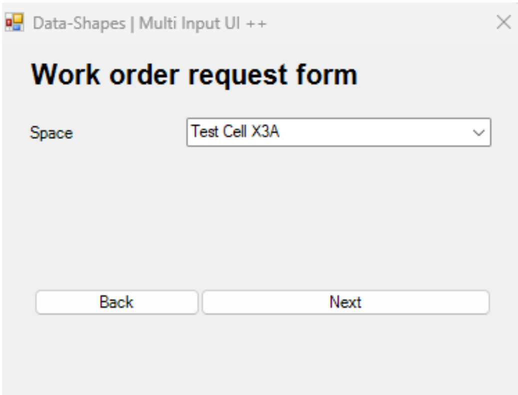


Figure E.1. Work order request module: selecting spaces.

Step 2: Select the symptom(s) from the checklist.

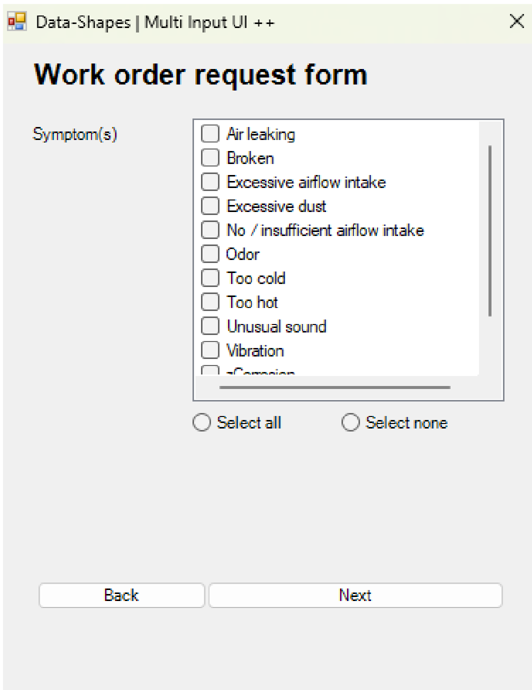


Figure E.2. Work order request module: selecting symptoms from the checklist

Step 3: Utilizing the symptom and space, filter potential fault assets and select the relevant assets in two ways:

(i) Select either from checklist

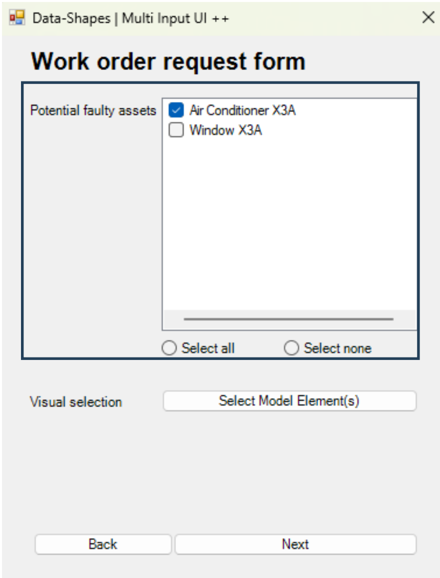


Figure E.3. Work order request module: selecting assets from checklist

(ii) Click on "Select Model Element(s)", it directly open 3D view of the space and visually select the relevant assets.

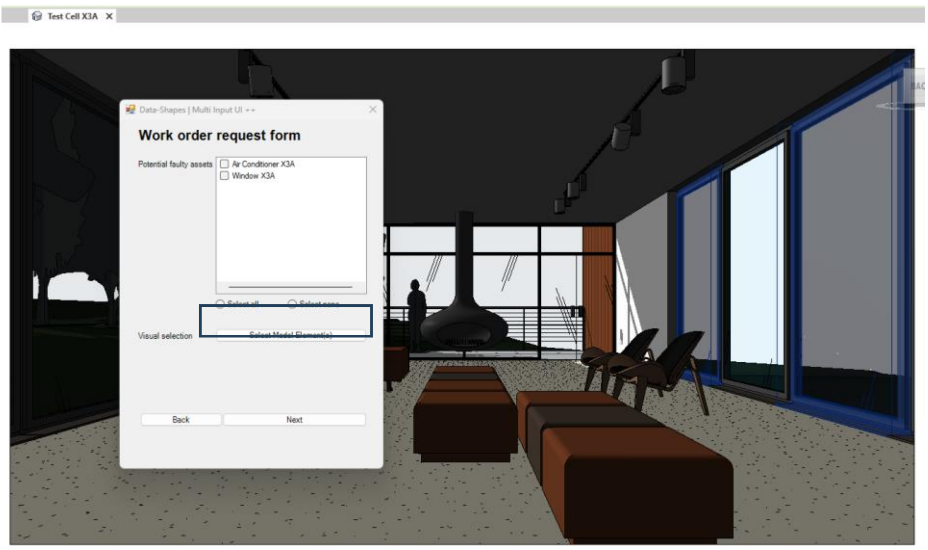


Figure E.4. Work order request module: visual selection of assets

Step 4: Add notes if available and finalize the request.

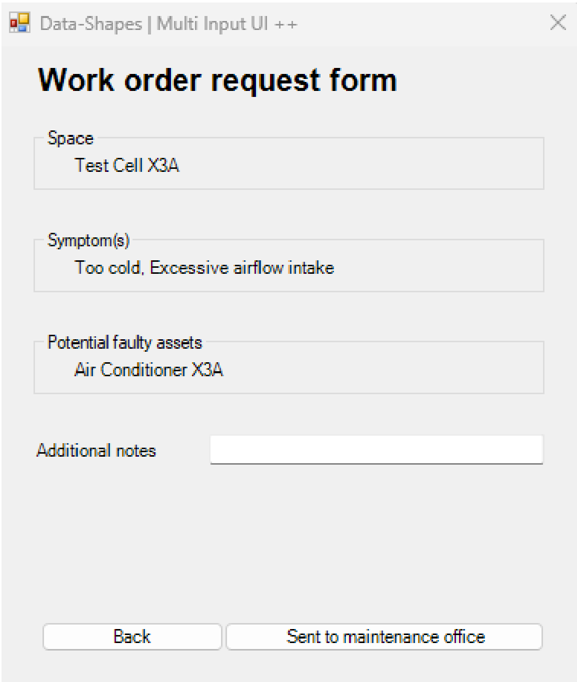


Figure E.5. Work order request module: request finalization

E.2. Site feedback module:

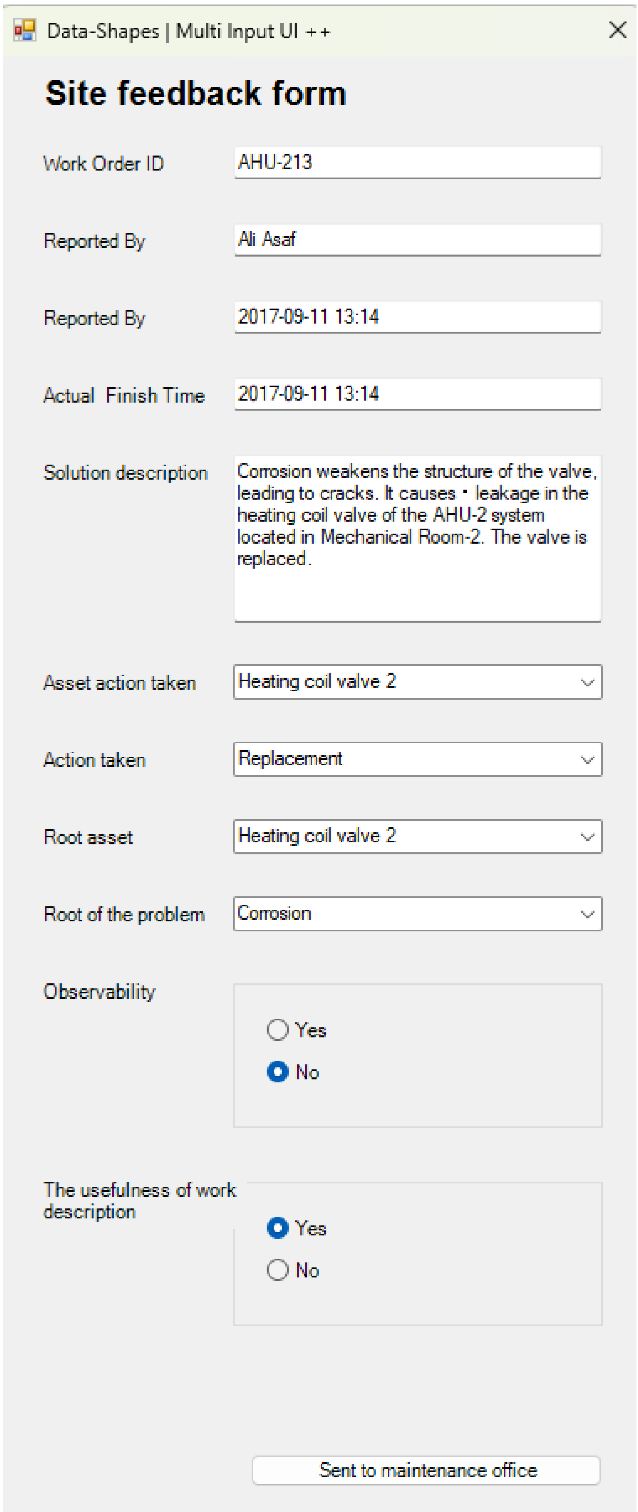


Figure E.6. Work order site feedback module

E.3. Work order evaluation and management module:

The solution description is updated by the facility maintenance office and integrated into BIM.

Description:

Corrosion weakens the structure of the valve, leading to cracks. It causes → leakage in the heating coil valve of the AHU-2 system located in Mechanical Room-2. The valve is replaced.

Utilizing the solution description, the fault network is filtered and drawn as follows:

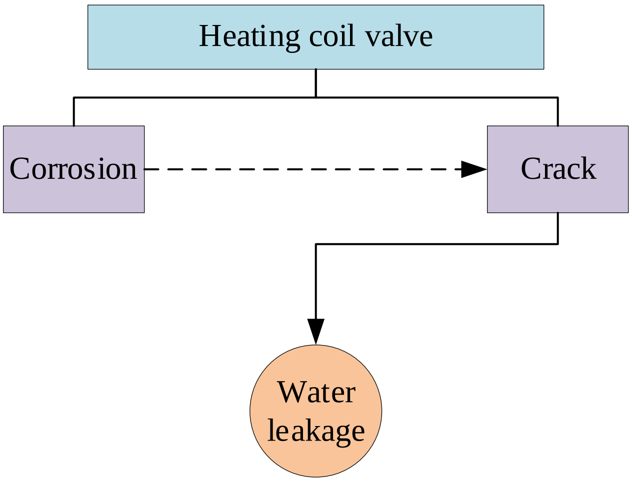


Figure E.7. Filtered fault network of example case

CURRICULUM VITAE

Surname, Name: Altun, Murat

EDUCATION

Degree	Institution	Year of Graduation
MS	METU Civil Engineering	2015
BS, Double Major	METU Industrial Engineering	2013
BS	METU Civil Engineering	2012
High School	Hacı Ömer Tarman Anatolian High School, Ankara	2007

FOREIGN LANGUAGES

Advanced English, Basic Arabic

HOBBIES

History, Math, Etymology, Board games

PUBLICATIONS

1. Altun,M. and Akcamete, A. “Yapay Arı Kolonisi Algoritmasının Zaman-Maliyet Ödünleşim Problemlerine Uygulanması”. 3. Proje ve Yapım Yönetimi Kongresi. 6-8 November, Antalya, 708-719 (2014).

2. Altun,M. and Akcamete, A.“BIM-based Multi-Objective Building Life Cycle Energy Performance Evaluation Using Particle Swarm Optimization”. 32nd CIB W78 Conference. 27-29 October, Eindhoven, The Netherlands,pp. 29-38,(2015).

3. Altun, M., Sonmez, R. and Akcamete, A. “Kaynak Dengeleme Problemine Öncelikli Diferansiyel Evrim Algoritmasıyla Bir Yaklaşım”. 4. Proje ve Yapım Yönetimi Kongresi. 3-5 November, Eskişehir, 293- 304 (2016).

4. Altun, M. and Akcamete, A. “Yapı Bilgi Modellemesi Tabanlı Bina Enerji Optimizasyonu”. 4. Proje ve Yapım Yönetimi Kongresi. 3-5 November, Eskişehir, 768-779 (2016).

5. Altun, M., Ersoz, A.B., Teke, T., Kurt, T., Akcamete, A. and Pekcan, O. “Application of Artificial Neural Networks on Building Energy Estimation.”, International Conference on Engineering Technologies (ICENTE'17), Konya, Turkey, (2017).

6. Altun, M. and Pekcan, O. “A modified approach to cross entropy method: Elitist stepped distribution algorithm”, Applied Soft Computing, Vol. 58, 756-769 (2017). doi: https://doi.org/10.1016/j.asoc.2017.04.032

7. Altun,M., Sonmez,R. and Akcamete, A. “A mixed integer programming method for multi-project resource leveling”, Journal of Construction Engineering, Management & Innovation, Vol 3 (2), 131- 140 (2018).
doi: https://doi.org/10.31462/jcemi.2020.02131140

8. Yalcin,Y., Altun,M. and Pekcan, O.“Analysis of Slopes Using Elitist Differential Evolution Algorithm”. International Conference on Engineering Optimization (EngOPT2018). 17-19 September, Lisboa, Portugal, pp. 815-826 (2018).

9. Altun, M. and Akcamete, A.” A Method for Facilitating 4D Modeling by Automating Task Information Generation and Mapping”, in a book entitled “Advances in Informatics and Computing in Civil and Construction Engineering” edited by Ivan Mutis and Timo Hartmann, Springer, Cham, pp. 479-486 (2019). doi: https://doi.org/10.1007/978-3-030-00220-6_57

10. Altun, M., Meral, C. and Akcamete, A. “Effect of envelope insulation on building heating energy requirement, cost and carbon footprint from a life cycle perspective”, Journal of The Faculty of Engineering and Architecture of Gazi University, Vol. 26 (6) ,1062- 1075 (2020).
doi: https://doi.org/10.17341/gazimmfd.445751

11. Altun, M., Meral, C. and Akcamete, A. “Investigation of the Effect of Outdoor Temperature on Building Heating Energy Requirement”, Pamukkale University Journal of Engineering Sciences, Vol. 35 (1), 147-164 (2020).
doi: https://doi.org/10.5505/pajes.2019.00334

12. Altun, M. and Pekcan,O. ” Optimum sizing of truss structures using a hybrid flower pollinations”, in a book entitled “Applications of Flower Pollination Algorithm and its Variants” edited by N. Dey, Springer, Singapore, 113-137 (2021).doi: https://doi.org/ 10.1007/978-981-33-6104-1_6

13. Tutus, E.B., Pekcan,O. Altun, M. and Turkezer, M. (2021).” Optimizing Reinforced Cantilever Retaining Walls Under Dynamic Loading Using Improved Flower Pollination Algorithm”, in a book entitled “Applications of Flower Pollination Algorithm and its Variants” edited by N. Dey, Springer, Singapore, 139-169 (2021). doi: https://doi.org/ 10.1007/978-981-33-6104-1_7

14. Altun, M., Yalcin, Y. and Pekcan,O..” A hybrid cuckoo search algorithm for cost optimization of mechanically stabilized earth walls”, in a book entitled “Applications of Cuckoo Search Algorithm and its Variants” edited by N. Dey, Springer, Singapore, 277-306 (2021). doi: https://doi.org/10.1007/978-981-15-5163-5_12

15. Altun, M. and Akcamete, A. “A hybrid intelligence framework for data-driven model-integrated fault resoning”, Journal of Construction Engineering and Management, submitted (2024).