A Graph-Based Method for Semi-Automating Clash Grouping and Resolution in BIM-Based Multidisciplinary Coordination

by

Farnaz KARIMI

THESIS PRESENTED TO ÉCOLE DE TECHNOLOGIE SUPÉRIEURE
IN PARTIAL FULFILLMENT OF A MASTER'S DEGREE
WITH THESIS IN CONSTRUCTION ENGINEERING
M.A.Sc.

MONTREAL, APRIL 23, 2026

ÉCOLE DE TECHNOLOGIE SUPÉRIEURE
UNIVERSITÉ DU QUÉBEC

BOARD OF EXAMINERS

THIS THESIS HAS BEEN EVALUATED

BY THE FOLLOWING BOARD OF EXAMINERS

Dr. Ivanka Iordanova, Thesis supervisor
Department of Construction Engineering, École de technologie supérieure

Dr. Ali Motamedi, Thesis Co-Supervisor
Department of Construction Engineering, École de technologie supérieure

Dr. Francis Bordeleau, President of the Board of Examiners
Department of Software Engineering and IT, École de technologie supérieure

Dr. Stéphane Morin-Pépin, Member of the Jury
Department of Construction Engineering, École de technologie supérieure

THIS THESIS WAS PRESENTED AND DEFENDED

IN THE PRESENCE OF A BOARD OF EXAMINERS AND THE PUBLIC

ON APRIL 16, 2026

AT ÉCOLE DE TECHNOLOGIE SUPÉRIEURE

ACKNOWLEDGEMENTS

First and foremost, I would like to express my deepest gratitude to my advisors, Prof. Ivanka Iordanova and Prof. Ali Motamedi, for their unwavering support, insightful guidance, and encouragement throughout my studies. Working with Prof. Iordanova has been a privilege; her thoughtful supervision and structured mentorship have significantly shaped my academic development. I am equally grateful to Prof. Motamedi, whose vision, critical insights, and constructive feedback have consistently refined and strengthened this research.

My sincere appreciation extends to all colleagues at the GRIDD Laboratory. I was fortunate to work in a supportive and collaborative environment, where collegiality and teamwork facilitated overcoming challenges.

I owe my deepest thanks to my family, especially my brother, for their unconditional love, patience, and steadfast support. Their belief has served as my primary motivation.

Une méthode fondée sur les graphes pour la semi-automatisation du regroupement et de la résolution des conflits en coordination multidisciplinaire BIM

Farnaz KARIMI

RÉSUMÉ

Une coordination multidisciplinaire efficace est essentielle pour éviter les conflits entre les composants du bâtiment. Bien que le Building Information Modeling (BIM) soit largement utilisé pour la détection automatisée des conflits, leur résolution demeure en grande partie manuelle et chronophage, ce qui limite l'efficacité des processus actuels de coordination basés sur le BIM.

Cette recherche propose une méthode visant à améliorer la gestion des conflits par l'intégration des données BIM et de la théorie des graphes. Le processus transforme les données de conflits en un réseau de composants et de relations, permettant d'analyser les conflits par groupes plutôt qu'individuellement. Grâce à cette approche, les éléments du bâtiment sont classés selon leur facilité de modification, les conflits connexes sont regroupés et des schémas de conflits récurrents sont identifiés. Cette démarche aboutit à une liste d'éléments pouvant être modifiés afin de résoudre les conflits au sein de chaque groupe.

Cette méthode aide les coordonnateurs à se concentrer sur les groupes de conflits les plus critiques et pourrait aider les modifications de conception redondantes en traitant collectivement les conflits liés. L'application de la méthode à des études de cas réels démontre son potentiel pour soutenir un processus de coordination BIM plus structuré et mieux informé. Les résultats montrent que la représentation des relations de conflits à l'aide de modèles de graphes améliore la compréhension des interdépendances entre les systèmes du bâtiment et favorise une approche plus systématique de la résolution d'interférences.

Mots-clés: modélisation des informations du bâtiment (BIM), coordination multidisciplinaire, gestion des conflits, théorie des graphes, résolution des conflits

A Graph-Based Method for Semi-Automating Clash Grouping and Resolution in BIM-Based Multidisciplinary Coordination

Farnaz KARIMI

ABSTRACT

Efficient multidisciplinary coordination is essential to avoid conflicts between building components. While Building Information Modeling (BIM) is widely used for automated clash detection, clash resolution remains largely manual and time-consuming, limiting the effectiveness of current BIM-based coordination workflows.

This research presents a method that makes clash management more effective by using both BIM data and graph theory. The process turns clash data into a network of components and their relationships, allowing clashes to be reviewed as groups instead of individually. With this approach, building elements are ranked by how easily they can be moved, related clashes are grouped, and repetitive conflict patterns are found. This results in a list of elements that can be moved to resolve clashes in each group.

This method can help coordinators focus on the most critical clash groups and reduces redundant design modifications by addressing related clashes collectively. The application of the method in real-world case studies demonstrates its potential to support a more structured and informed BIM-based coordination process. The results show that representing clash relationships using graph models improves the understanding of interdependencies between building systems and supports a more systematic approach to clash resolution.

Keywords: building information modeling (BIM); multidisciplinary coordination, clash management, graph theory, clash resolution

TABLE OF CONTENTS

Page

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

3D	Three Dimensional
AEC	Architecture, Engineering, and Construction
AI	Artificial Intelligence
BCF	BIM Collaboration Format
BIM	Building Information Modelling
BIMIMC	BIM Model Management Control
DSR	Design Science Research
GUI	Graphic User Interface
HVAC	Heating, Ventilation, and Air Conditioning
IFC	Industry Foundation Classes
MEP	Mechanical, Electrical, and Plumbing
MVC	Minimum Vertex Cover
MWVC	Minimum Weighted Vertex Cover
SCOP	Sequential Composite Overlap Process
SLR	Systematic Literature Review
VC	Vertex Cover

INTRODUCTION

Effective design coordination is essential for the success of construction projects, as it addresses potential system conflicts before construction begins (Wang & Leite, 2016a). Proper management of design helps prevent budget overruns, delays, and risks to structural safety, ensuring that designers clearly understand their roles and how their work intersects with other fields (Leite, 2019). The significance of this coordination becomes even more critical in complex structures, such as hospitals, where various systems are installed by different stakeholders in confined spaces.

Clash management primarily consists of detecting and resolving clashes. Traditionally, design coordination was achieved by overlaying 2D drawings on a light table, using a method known as the sequential composite overlap process (SCOP) (Khanzode, Fischer & Reed, 2008). However, this traditional approach was labor-intensive and error-prone, often resulting in overlooked clashes (Wang & Leite, 2012). The adoption of Building Information Modeling (BIM) has significantly altered clash management, introducing technological advancements that impact both organizational and procedural levels.

Various strategies have been developed to address these challenges, with BIM emerging as the most effective (Cao *et al.*, 2015). BIM tools facilitate the sharing and integration of information across multiple disciplines, thereby enhancing collaboration and coordination throughout the project stages. A building involves thousands of components, and a fundamental collaborative task is to manage the layout and dimensions of these components to avoid design clashes. Research indicates that 80-90% of construction project failures are due to design errors (Love, Edwards, Smith & Walker, 2009). Consequently, proficient design coordination and clash management are vital for the success of a project. BIM has been increasingly utilized for these purposes, with numerous studies noting that design coordination and clash detection are the most beneficial and frequently used features (Cao *et al.*, 2015; Mehrbod, Staub-French, Mahyar & Tory, 2019a).

As shown in figure 0.1 in a BIM-enabled construction project, the coordination activity includes integrating 3D BIM models from different disciplines using software such as Navisworks. Before this step, the different disciplines must check their own models to make sure they are free of clashes. Once the models are combined into a federated model, the first clash detection is conducted to identify any physical or spatial conflicts between systems. Following the initial clash detection, the BIM coordinator reviews the 3D federated model to identify any soft clashes issues where insufficient space is left for maintenance or operation. The detected clashes are then manually grouped. These grouped clashes are then discussed collectively during design coordination meetings. After all hard and soft clashes are sorted and classified, a coordination meeting is held to discuss solutions. The different disciplines are responsible for resolving the clashes within their respective scopes. If any clashes remain unresolved, the process is repeated until all issues are addressed. Once all clashes are resolved, the project can progress smoothly toward completion. This BIM coordination workflow was developed by observing industry practices and refined through a review of the relevant literature. However there are limitations observed in this approach such as:

- Despite technological advances, grouping related clashes remains a manual process, and in many cases, the dependencies between clashes are not considered, so they are discussed and resolved individually.
- Clash resolution is still a manual and time-consuming task.

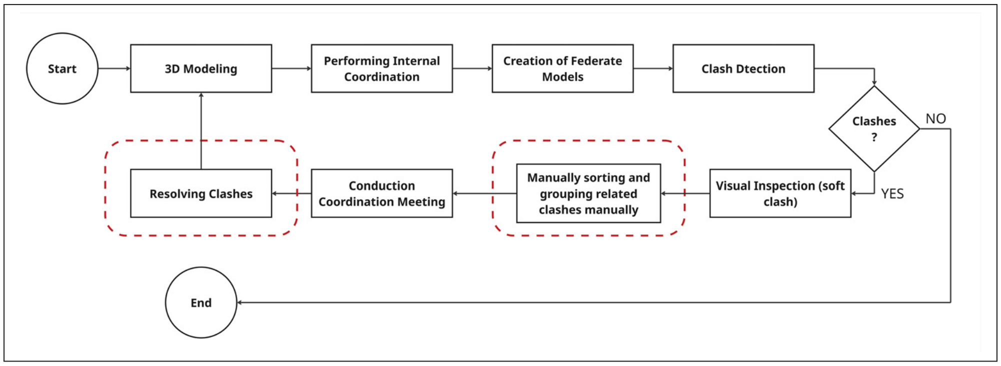


Figure 0.1 Design coordination workflow, with the areas covered by this thesis highlighted by the red dashed rectangles.

This thesis is part of a larger research project on automating clash management (Meem & Iordanova, 2022a,b) and focuses on the automation of clash grouping, the identification of similar clash clusters, and clash resolution. The area covered by this thesis is highlighted within the red dashed rectangle in figure 0.1.

Based on the issues outlined above, the central research question is formulated as follows:

How can clash grouping and resolution be improved in multidisciplinary BIM coordination to reduce rework and improve the efficiency in clash management avtivity?

Research goals

This research aims to propose a method to enhance multidisciplinary coordination by optimizing the clash resolution process within BIM-based workflows. To address this primary aim, several specific objectives have been identified and are summarized below:

- Identify gaps in current BIM-based clash management, focusing on limitations in clash grouping and clash resolution.
- Develop a method to improve the management of detected clashes in BIM-based multidisciplinary coordination.

Structure of the thesis

This thesis consists of five chapters. Chapter 1 reviews the background, history, current industry practices, and recent developments in BIM-based multidisciplinary coordination and clash management. Chapter 2 details the research methodology. Chapter 3 presents the main solutions developed. Chapter 4 describes the application of the methodology in two real-world case studies, including results, comparative analyses, and key insights. Chapter 5 discusses the findings in relation to the research questions and existing literature, highlights theoretical and practical implications, addresses study limitations, and outlines directions for future research. Finally, a summary of proposed solutions, key findings, and main contributions of the research is provided.

CHAPTER 1

LITERATURE REVIEW

This chapter reviews existing research on clash management in BIM-based multidisciplinary coordination. It synthesizes current approaches, highlights their limitations, and identifies research gaps that motivate the development of the proposed system.

1.1 Introduction

Clash management, which is a way to avoid mistakes in the early phases of construction and ensure the quality of the design, plays a vital role in achieving success in a project. However, unforeseen cost hikes and construction delays still happen in projects. Clash management consists of two processes: clash detection and clash resolution. BIM based tools have been developed over the years for detecting clashes, but the clash resolution process is still manual and time-consuming. Statistics indicate that design errors account for 80–90% of failures in construction projects (Hu & Castro-Lacouture, 2019). Clash resolution in multidisciplinary coordination refers to identifying and resolving conflicts or inconsistencies between design and construction elements in a building project (Lee *et al.*, 2014). The design and construction of a building involve a range of professionals with diverse areas of expertise, such as architects, engineers, and contractors. These professionals are responsible for creating and executing the design of the building, which includes various components such as structural, electrical, plumbing, and HVAC systems (Lee *et al.*, 2014). In order to ensure that these components function seamlessly, it is crucial to identify and resolve any clashes between them. To overcome this problem, many approaches are proposed from various perspectives. In addition, clash identification involves issues, but it does not involve resolving these clashes (Pärn, Edwards & Sing, 2018). BIM in construction projects enhances collaboration between different disciplines by providing a shared digital platform for communication and coordination (Pärn *et al.*, 2018; Ciribini, Ventura & Paneroni, 2016). BIM also increases accuracy in the construction design, reduces the risk of errors, and enables automatic clash detection. BIM saves cost and time by reducing the need for rework and minimizing the time required to resolve clashes among multidiscipline (Eastman, 2011).

Additionally, Artificial Intelligence (AI) has been increasingly applied in clash resolution in multidisciplinary coordination. The application of AI in this area can reduce the time and cost associated with the manual resolution of clashes. For example, AI algorithms can evaluate alternative design solutions and identify the most cost-efficient solution. This approach can reduce the time and cost associated with manual analysis and decision-making and improve the overall design quality (Harode & Thabet, 2021). This literature review aims to study the proposed clash resolution strategies for efficient multidisciplinary coordination, identify gaps in current knowledge or research, and identify areas for future research.

1.2 Methodology

The methodology adopted for this literature review is a Systematic Literature Review (SLR). SLR is a comprehensive, systematic, and transparent method for identifying, evaluating, and synthesizing existing research on a specific topic (Kitchenham *et al.*, 2004). It is an evidence-based approach used to gather and critically analyze information from published literature to provide a thorough and up-to-date overview of a particular area of research. The end goal of a systematic literature review is to provide an objective, accurate, and comprehensive summary of the current state of knowledge on a particular topic. This review can assist in informing decision-making, guide future research, and support the development of new theories and practices (Tranfield, Denyer & Smart, 2003). This study utilizes quantitative and qualitative analyses to investigate existing literature on clash resolution in multi-disciplinary coordination. This literature review was conducted in eight steps, see figure 1.1.

1.2.1 Defining Research Questions

The following research questions were established to guide the SLR:

1. What are the most common approaches to resolving clashes in multidisciplinary coordination projects, and how effective are they in practice?

   The goal is to identify the methods utilized for resolving clashes, understand their implementation and assessment, and uncover the difficulties encountered in their application.

2. What are the best practices for incorporating lessons from previous clash resolution experiences into future projects to improve the clash resolution process in multidisciplinary coordination?

   The goal is to assess the current shortcomings and establish potential areas for future research based on the analyzed literature and findings from question 1.

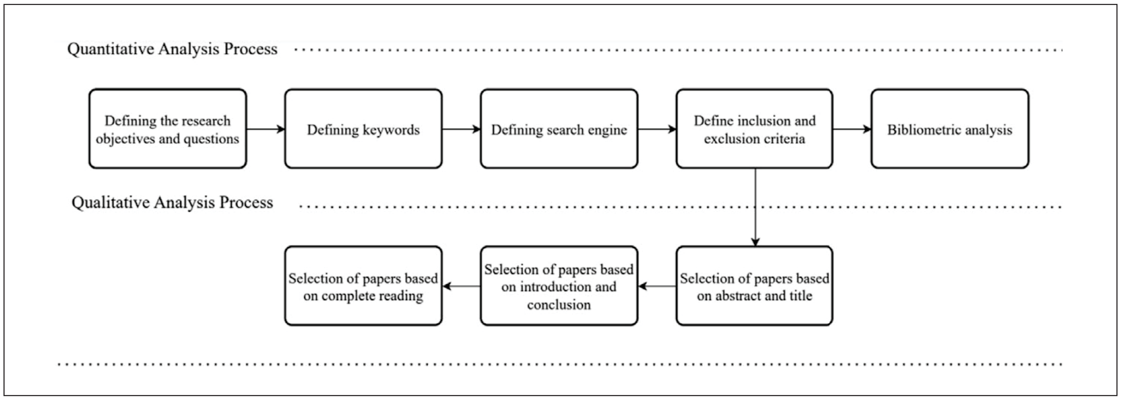


Figure 1.1 Methodology workflow

1.2.2 Defining Keywords

To identify articles related to Clash resolution in multidisciplinary coordination, these keywords were picked due to their relevance to the topic under examination: (“Building information model*” OR BIM) AND (“automat*” OR “Artificial intelligence” OR “Machine learning” OR AI) AND (clash OR interference OR overlap* OR “clash resolution”).

1.2.3 Defining Search Engine

To thoroughly examine the literature, we utilize the Scopus database resources. The Scopus database offers a comprehensive collection of STM (Science, Technology, and Medicine) journal articles and their corresponding references, facilitating bi-directional searches across time. This vast database can be a valuable tool for collection development and research purposes (Burnham, 2006).

1.2.4 Defining Inclusion and Exclusion Criteria

Based on the selected keywords, 186 publications were initially identified from the database. The review considered journal articles and conference papers published in English. After applying the scope and coverage criteria, including time frame, language, and source type, the number of relevant publications was reduced to 147. Then a qualitative analysis was conducted by reading all of the selected articles, leading to 15 articles being included in the qualitative analysis (figure 1.2).

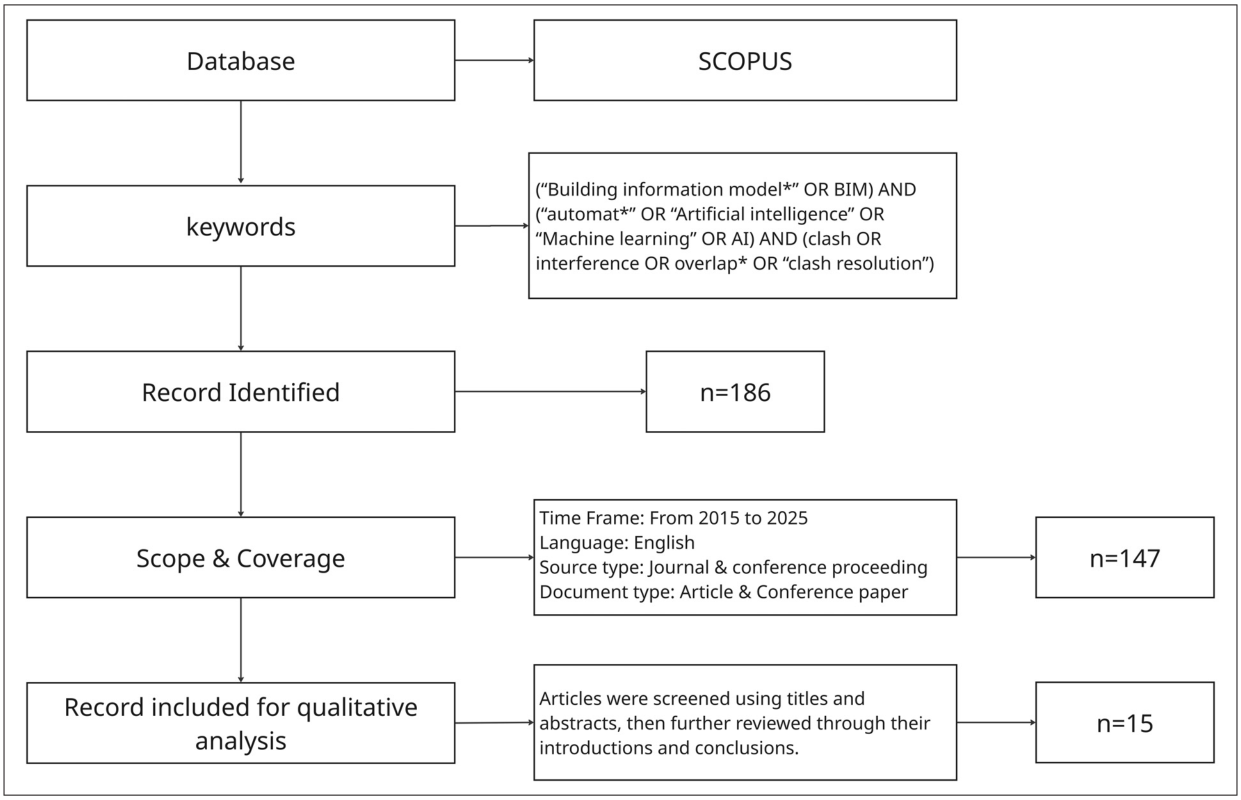


Figure 1.2 Article selection process

1.2.5 Bibliometric Analysis

This study took into account a 10-year time frame spanning from 2015 to 2025. Figure 1.3 shows the number of publications per year among the 147 articles related to the studied topic.

Figure 1.3 shows that the number of publications has generally increased over time, even though there were some ups and downs in the early years. Publication activity was relatively low from 2015 to 2018, but it started to rise in 2019 and stayed higher in the following years. The period from 2023 to 2025 stands out for its strong growth, which suggests that research interest in this field is increasing.

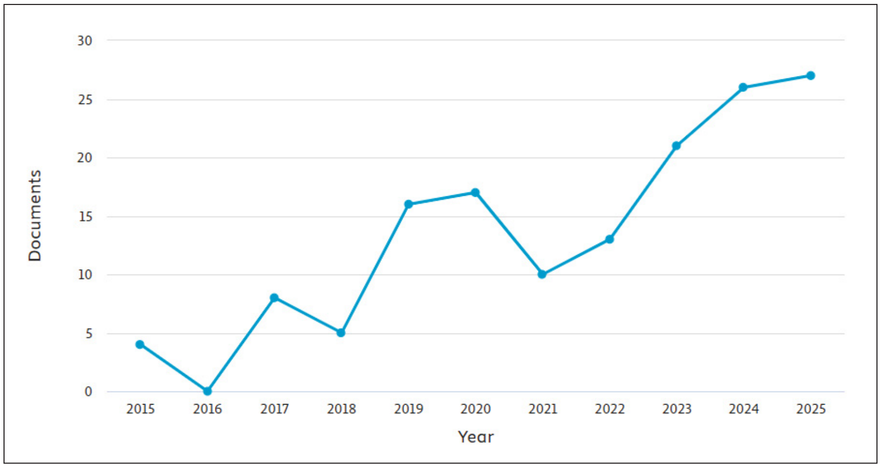


Figure 1.3   Articles published per year

To gain a deeper understanding of the relationships between the keywords, we utilized the Scopus and VOSviewer software to create networks of subdomain research. In this research, we analyzed the keywords used in the research publications related to the topic of interest. This study utilized a co-occurrence network map to investigate the interrelation-ships between keywords and discern essential research topics within the domain of BIM-based multidisciplinary coordination. As illustrated in figure 1.4 the bibliometric network highlights several major research clusters related to BIM, including architectural design, construction, automation, and data-driven methods. Architectural design appears as a central theme, connected to BIM and clash-related topics. While concepts such as automation, machine learning, and artificial intelligence are present, clash resolution itself remains a relatively limited and less consolidated research focus, indicating the need for further dedicated studies in this area.

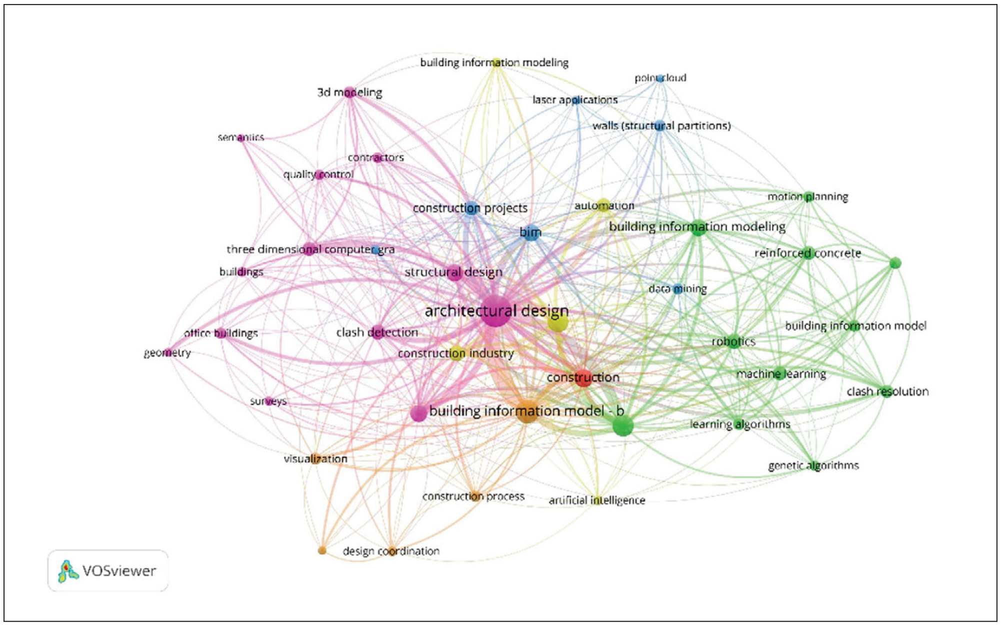


Figure 1.4 The keyword co-occurrence network map highlights the most significant research areas

1.3 Qualitative Analysis

We conducted a systematic literature review to examine the contributions to clash resolution in construction projects thoroughly. The qualitative analysis was conducted on 15 selected articles, as illustrated in figure 1.1.

The articles were classified manually and based on their subjects. The categories are: Acquiring clash information, clash classification and clash resolution methods. This categorization provides a clear distinction between the various subdomains within the field. Figure 1.5 illustrates an overview of these three categories.

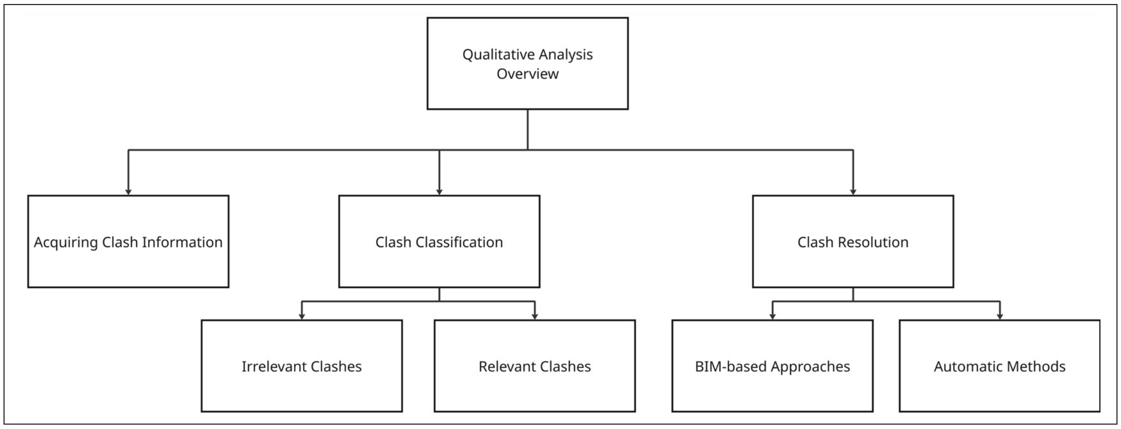


Figure 1.5 Qualitative analysis overview

1.3.1 Acquiring Clash Information

Despite the importance of BIM coordination, there is often a lack of formalized methods for collecting and storing clash-related information generated during the process. This can make it difficult to learn from past projects and apply those lessons to future endeavors, ultimately limiting the effectiveness of MEP coordination efforts (Wang & Leite, 2016a,b).

Wang & Leite (2016a,b), proposed a formalized schema that captures clash features and solutions during BIM coordination, integrating findings from literature reviews, field studies, and a laboratory experiment. The schema provides a structured approach for documenting clashes and managing coordination, including object-based and clash-based information. The information like clash ID, location of clash, discipline, component name, and how engineers resolved the clash is presented in this study. Korman, Fischer & Tatum (2003) present a structured framework that takes into account the fundamental information and logical procedures that are necessary to carry out BIM coordination. Assembling, evaluating, and illustrating the required data for BIM coordination led to the creation of a framework that caters to the specific requirements of stakeholders involved in this crucial procedure. This group is also associated with clash resolution as a situation may arise where a clash is encountered repeatedly in multiple projects. By leveraging the knowledge gained from previous projects and lessons learned, the resolution process can be expedited, saving time in resolving such clashes.

1.3.2 Clash Classification

Hu & Castro-Lacouture (2019) use both rule-based reasoning systems and machine learning classifiers to classify BIM-detected conflicts. Initially, data such as clash points and distances were gathered. Subsequently, a set of clashes was manually labeled and categorized into four groups, namely severe clashes which are the clashes that will affect the project and must be resolved (hard clashes), negligible clashes (soft clashes), 'legal interventions' indicate intentional clashes originating from the designer, such as the pipes and conduits penetrating through the slabs, and unknown clashes. Following this, a combination of rule-based reasoning and single and multiple machine learning classifiers were employed to automatically classify the rest of the clashes. After this article, the authors proposed a development method that that utilizes a combination of rule-based reasoning and supervised machine learning to automatically detect and eliminate irrelevant clashes (Lin & Huang, 2019). Hu & Castro-Lacouture (2019) explore several algorithms, namely J48-based decision tree, random forest, Jrip-based rule methods, binary logistic regression, naïve Bayes, and Bayesian network to examine which method is more accurate for clash classification. Moreover, a novel approach is presented to identify irrelevant clashes, which illustrates how historical data can be leveraged to enhance the clash management process. However, the model's effectiveness is limited by the level of automation employed in the data collection process and the size of the dataset. Despite achieving a prediction accuracy of approximately 80%, some relevant clashes still need to be correctly identified as irrelevant. This misclassification poses a potential hindrance to project outcomes.

1.3.3 Clash Resolution

Once information about clashes is captured, categorized into different classes, and irrelevant clashes are filtered out, the next step is to resolve the remaining clashes.

In this study clash resolution approaches can be classified into two categories: (1) BIM-based coordination approaches, (2) automated approaches. The following sections review these two categories.

- **BIM-Based Approaches**:

  BIM-based clash resolution method is a technique used to resolve conflicts between different components in a construction project using BIM. In this approach, the coordination process is entirely manual.

  Lee *et al.* (2014) compares two BIM-assisted MEP coordination strategies on the same pharmaceutical office building, where two coordinators used different methods in separate zones. One team used a parallel approach, modeling all trades at once and checking for clashes later. The other team used a sequential approach, coordinating trades one after another in a set order. The authors found that the sequential strategy was about three times faster and led to fewer clashes early on, which helped shorten meetings and reduce the number of coordination cycles. They suggest this is mainly because of how information was shared: the parallel method kept decisions and information with the coordinator, while the sequential method gradually shared coordinated models with everyone, making later modeling easier and cutting down on rework. However, the study has some limitations. It is based on just one case study of a single building, and the results also depend on the coordinators' experience, so the productivity gains may not be due only to the coordination strategy.

  In order to enhance the application of BIM in MEP systems, Wang, Wang, Shou, Chong & Guo (2016) have created a practical BIM framework that facilitates the coordination of MEP layout from the design stage to the construction stage. This framework consists of five distinct levels of detail for BIM models, which include a 3D MEP preliminary design model, 3D MEP detailed design model, 3D MEP construction design model, MEP construction model, and MEP prefabrication model. Adhering to this framework provides project teams with an effective means of managing and coordinating MEP systems throughout the entire project lifecycle. This leads to improved project efficiency and reduced likelihood of errors or rework. The limitation of this study is that the case study primarily focused on the perspective of the BIM consultant and needed more detailed information on third-party cost savings.

- **Automated Methods**:

  Automatic clash resolution methods refer to computational approaches that use predefined rules, heuristics, or optimization algorithms to resolve conflicts in construction projects.

Hsu & Wu (2019) proposed a framework utilizing the API provided by Revit; the study employs a simulated annealing algorithm to identify layout modifications that minimize the occurrence of design conflicts. The proposed method has five steps:

1. Randomly choose a clash.
2. Decide which object to modify.
3. Revise the chosen object.
4. Modify the object and create a clash list.
5. Record the number of clashes after modifying the object .

The paper limitation is not mentioned in the manuscript. As the authors use a list of priority of the object changed, if we have other types of MEP objects (for example, fuel gas piping—the number of objects is high) in a building, we cannot apply the same result to that building as we do not know what the priority of these other objects is. Therefore, we do not know how to resolve those clashes, which is the limitation.

Graphs effectively illustrate the relationships between different parts of a system. Unlike tables or lists, they represent both components and their connections, offering a clearer understanding of system interactions. Graph-based representations also support advanced analysis methods (Das & Soylu, 2023). For example, network algorithms can identify important groups, determine shortest paths, or highlight key nodes within a system. These capabilities make graphs particularly valuable for decision-making in complex environments where understanding both components and their interdependencies is essential (Newman, 2010).

Several studies have explored graph-based methods in construction and BIM research. Chinowsky, Taylor & Di Marco (2011) developed coordination networks to analyze task dependencies and organizational performance in construction projects. Han, Lee & Peña-Mora (2012) highlighted the importance of contextual clash information and demonstrated the potential of graph-based learning techniques for predicting component changes during clash resolution.

Hu, Castro-Lacouture, Eastman & Navathe (2020b), introduced a graph-theoretic framework that optimizes clash resolution sequences by modeling the problem as a minimum feedback arc set. By integrating graph theory with BIM data, their approach clarifies interdependencies among clashes and enables optimization of correction sequences through parallel and sequential strategies. The optimized sequence also groups related clashes, allowing project participants to address them collectively.

Despite these contributions, current graph-based BIM studies have several limitations. Most approaches rely heavily on BIM data quality and are often validated only with limited or simplified case studies. Many methods use assumptions that do not fully reflect real-world coordination constraints. Additionally, integrating graph-based techniques into practical BIM coordination workflows remains limited, so clash grouping and resolution are still largely manual.

Recent studies have sought to reduce reliance on handcrafted rules by integrating machine learning with graph representations. Hu, Xia, Chen & Gao (2023) applied graph convolutional networks to learn change-component decisions directly from component dependency graphs, demonstrating improved prediction accuracy by capturing clash context automatically. Despite these advances, the approach requires extensive data preparation and does not fully address temporal evolution across design stages.

Harode, Thabet & Gao (2022) suggest implementing a hybrid supervised and reinforcement learning model to automate conflict resolution. The model incorporates supervised learning to develop an initial probabilistic model that predicts conflict resolution decisions based on the given information. The reinforcement learning component then uses this probabilistic model as a starting point and continuously refines its accuracy through interaction and feedback.

1.4 Conclusion

This chapter provided a systematic literature review on the use of BIM and automated approaches in the process of clash management in multidisciplinary coordination in construction projects. The review was based on a search of 147 articles published in journals and conference proceedings indexed in Scopus. The aim of the review was to identify the most common approaches to resolving clashes and the best practices for incorporating lessons learned from previous experiences into future projects. The results show that BIM is a powerful tool for supporting multidisciplinary coordination, while automate approaches demonstrate potential for reducing the time and cost associated with manual clash resolution.

The reviewed studies distinguish between BIM-based and automatic approaches to clash resolution. BIM-based approaches rely primarily on BIM tools and human decision-making, whereas automatic approaches employ automated techniques to support or automate parts of the resolution process. Although automated methods show promising results in terms of efficiency, the literature indicates that these approaches still require further development to address practical limitations and ensure broader applicability in real coordination workflows.

The literature review shows that most existing manual and automatic approaches address clashes individually and do not consider dependencies between building components. This gap underscores the need for methods that can systematically represent and analyze relationships among components and clashes. Graph-based representations offer a promising solution by explicitly modeling these relationships and interdependencies. Therefore, this research adopts a graph-based approach to address current limitations, including the isolated treatment of clashes, the limited consideration of component interdependencies, and the lack of systematic methods for grouping clashes and minimizing modification effort in real coordination workflows.

CHAPTER 2

RESEARCH OBJECTIVES AND METHODOLOGY

This chapter presents the research objectives, questions, and hypotheses developed to address the gaps identified in the previous chapter. It also details process of the design, development, and evaluation of a structured solution to improve clash grouping and resolution in multidisciplinary BIM coordination.

2.1 Introduction

Design Science Research (DSR) was adopted as the principal methodology for this study due to its suitability in addressing complex, real-world problems through the development of innovative and practical solutions (Hevner, March, Park & Ram, 2004). Unlike natural sciences, which aim to explain and understand reality, DSR is a problem-solving paradigm that emphasizes the creation and evaluation of artifacts such as constructs, models, methods, and systems that serve specific purposes (March & Smith, 1995; Vom Brocke *et al.*, 2020). In doing so, DSR not only generates effective solutions but also contributes to the advancement of human knowledge by producing design knowledge (Hevner *et al.*, 2004).

Peffers, Tuunanen, Rothenberger & Chatterjee (2007) formalized DSR into a process model with three overarching objectives: building on prior research in design science, offering a structured framework to guide researchers, and providing a mental model for evaluating and presenting design outputs. They also outlined six key activities that shape the methodology: problem identification and motivation, defining the objectives of a solution, design and development, demonstration, evaluation, and communication. These stages ensure a systematic and iterative approach that fosters critical reflection and the continuous refinement of solutions.

Furthermore, as emphasized by De Sordi (2021), the DSR approach is characterized by iterative design cycles in which theoretical insights are combined with practice to progressively enhance proposed solutions. This integration of academic rigor with industrial applicability makes DSR particularly valuable in contexts where concrete, sector-specific solutions are required.

Accordingly, in this research, DSR was employed not only to theorize and investigate the concept of built asset lifecycle information coupling but also to design and validate practical artifacts that directly address the needs of both academia and industry.

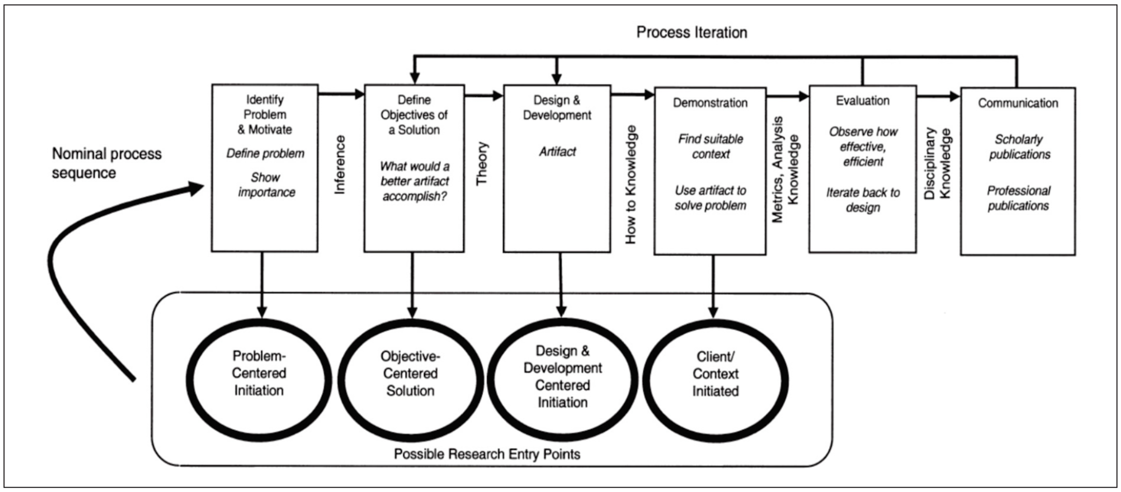


Figure 2.1 DSR methodology process model
Taken from (Vom Brocke *et al.*, 2020).

2.2 Motivation and problem statement

2.2.1 Motivation

Clash management is essential for maintaining design quality and minimizing costly change orders during construction (Mehrbod, Staub-French, Mahyar & Tory, 2019b). As building systems become more complex, detecting and resolving clashes across multiple disciplines poses significant challenges. While BIM has enhanced the efficiency of clash detection, current BIM-enabled coordination is limited by a primary issue: clash resolution remains a manual, fragmented process that addresses conflicts individually, without leveraging component interdependencies.

Observations from BIM coordination meetings show that decisions about changing components and the sequence of fixing clashes are rarely made effectively. In addition, earlier studies have mostly overlooked how components depend on one another (Korman & Tatum, 2001; Korman *et al.*, 2003; Radke, Wallmark & Tseng, 2009). This study propose to optimize clash resolution using a graph-based approach and prioritize component clashes based on their type.

2.2.2 Problem statement

BIM-enabled clash detection primarily identifies conflicts among building elements but does not inherently provide solutions for their resolution. Currently, the role of BIM in clash resolution remains limited. During coordination meetings, engineers manually analyze BIM models to interpret clash contexts by rotating views, measuring distances, and discussing potential solutions in relation to design, construction, and maintenance constraints (Mehrbod *et al.*, 2019b). Consequently, the function of BIM models in resolving clashes is similar to that of traditional 2D drawings, offering visual clarification rather than analytical reasoning.

While BIM enables the simultaneous detection of multiple clashes and offers the potential for holistic resolution, current practice continues to address clashes individually without accounting for their interdependencies (Korman *et al.*, 2003; Wang & Leite, 2013). Previous studies have enhanced detection algorithms (Beetz, van Berlo, de Laat & van den Helm, 2010) but have not effectively incorporated construction context or reduced coordination workload (Hu, Castro-Lacouture & Eastman, 2019). Furthermore, clashes are ill-structured problems that are context-dependent and iterative (Jonassen, 1997). Modifying one element can generate new clashes elsewhere due to spatial dependencies (Tabesh & Staub-French, 2005). Thus, existing BIM-based clash management remains fragmented and inefficient, lacking a systematic approach that captures inter-component dependencies to support holistic and optimized clash resolution.

2.3 Research objectives, questions and hypothesis

2.3.1 Main research objective

The primary objective of this study is to enhance the efficiency of clash management in BIM by addressing both clash grouping and providing assistance for resolving similar issues by identifying them. In terms of grouping, the research aims to develop a method capable of automatically grouping related clashes and then clustring similar clash groups based on element types. This approach enables a more structured and meaningful organization of clashes, reducing redundancy and improving coordination. Regarding resolution, the study seeks to introduce a novel approach that advances current practices by prioritizing building components to resolve clashes, minimizing modifications while maximizing effectiveness.

2.3.2 Research questions

To fulfill the goal of this research, the detailed questions addressed by the study are as follows:

1. How can dependencies among building components be effectively considered to enable automatic grouping and the identification of similar groups that share nearly the same solution?
2. What strategies can be developed to prioritize building components in the clash resolution process, ensuring more efficient and practical decision-making?
3. How can the proposed strategy be implemented within a BIM-based coordination workflow to support practical application in real projects?

2.3.3 Research Hypotheses

1. Automatically grouping related clashes based on element types enables the identification of clash groups that share common resolution strategies, thereby reducing repetitive review of similar clashes.
2. Prioritizing building components based on their relative importance and flexibility leads to more effective clash resolution strategies by reducing coordination time and minimizing disagreements among disciplines over which components should be modified.
3. The integration of automated clash grouping and resolution provides a systematic and practical framework that enhances the overall efficiency and reliability of BIM coordination processes.

2.4 Artifact design and development

The design and development process in this research project is systematically organized using the DSR approach. The DSR methodology emphasizes addressing specific needs, ensuring simplicity in execution, integrating components effectively, and fostering continuous innovation. This approach follows an iterative cycle, beginning with conceptual modeling and progressing to more detailed designs, enabling ongoing adjustments. According to De Sordi (2021), employing an instantiation artifact in research facilitates practical analysis of the proposed solution. Such practical evaluation is essential, as it determines whether the developed artifact effectively addresses the identified problem. Implementation of the artifact enables direct observation of its performance and assessment of its impact in real-world contexts, thereby providing concrete evidence of its efficiency and suitability for the initial requirements.

2.5 Evaluation

According to DSR methodology, naturalistic evaluation requires that the developed artifact be applied under real conditions, by real users, and in response to real problems. It is therefore essential that the artifact be tested within the same context for which it was designed (De Sordi, 2021).

The developed artifacts generate results that address the research objectives and answer the research questions. Within the adopted design science methodology, evaluation of these results occurs in the final step. Accordingly, the developed artifact is assessed through a field experiment conducted as a case study. Further details are provided in Chapter 4.

2.6 Research communication

In the final phase of DSR methodology, effective communication of the research outcomes is essential. As emphasized by De Sordi (2021), this phase focuses on disseminating the knowledge generated through the design, development, and evaluation of the artifact to both academic and professional audiences. In this research, the thesis itself constitutes the communication artifact, as it documents the research problem, methodological choices, artifact development, case study evaluation, and results.

CHAPTER 3

PROPOSED SYSTEM FOR CLASH MANAGEMENT IMPROVEMENT

This chapter describes the proposed clash management method and explains how it is implemented in practice. It outlines each workflow step, including defining element priorities, detecting clashes, grouping conflicts, and identifying optimal resolution actions.

3.1 Observed coordination workflow in practice

Before designing the proposed system, we met with five experienced BIM coordinators. This discussion clarified current clash detection and resolution practices and helped identify areas for improvement. The coordinators outlined a systematic and flexible workflow based on the principle “organize first, resolve later”. Detected clashes are first grouped by element type, such as walls, ceilings, or ventilation systems, to support visualization and management. If a category becomes too large or complex, it is divided into smaller subgroups for detailed review. The building under coordination is divided into distinct sectors typically per level to simplify navigation and manage spatial complexity. During sector analyses, the coordinator keeps the entire model visible to determine the extent of each system and assess its interactions with other building components. The coordination process operates on a weekly cycle. Regular clash tests are conducted, results are published for review, and coordination meetings are held to address issues. Early cycles focus on main systems and large distribution lines, while later cycles address details such as smaller components. Meetings are organized by sector or discipline to ensure efficiency and focus. The coordinators note that MEP coordination is highly context-dependent. Slope requirements, vertical layering constraints, and limited ceiling space often determine the feasibility of solutions. As a result, complete automation is not yet possible; human judgment is still required to assess constructability and design intent. However, the coordinators believe partial automation could improve efficiency, especially with tools that group clashes by element type and algorithms that detect repeated patterns in similar layouts. These capabilities would allow one validated solution to be applied consistently across identical conditions. The coordination process observed can be summarized in six iterative steps:

- Run clash tests and collect newly identified conflicts.
- Group clashes by element type and assign them to existing validation categories or create new groups as needed.
- Conduct full-context visual inspections to identify false positives (e.g., "duct vs. room").
- Isolate valid clashes into smaller working groups and mark them as Active.
- Publish grouped issues through a BIM Collaboration Format (BCF) management platform for review and resolution by the responsible disciplines.
- Repeat the process weekly, with meetings organized by sector or system type.

Insights gathered from these discussions informed the design considerations of the proposed graph-based grouping framework presented in the next section. The framework aims to automate repetitive aspects of clash grouping while preserving human oversight and contextual decision-making, consistent with observed coordination practices.

3.2 Proposed workflow for improving clash management

This study proposes an artifact (A graph-based clash management methodology) implemented within a BIM coordination workflow. This approach aims to enhance the BIM coordination process and reduce redundant work. The artifact pursues three goals: first, to categorize clashes based on their characteristics, such as types of building components; second, to cluster similar groups of clashes; and third, to identify the minimum number of components to modify in order to minimize changes and streamline the resolution process.

Figure 3.1 presents the high-level process of the proposed artifact, illustrating its two main operational layers: an offline setup phase performed once per project, and a run-time process executed during each coordination cycle.

As illustrated in Figure 3.1, the offline process, performed once per project, consists of the Elements Priority Identification step, in which each building component is assigned a priority value reflecting how easily it can be modified. This priority structure is established before any

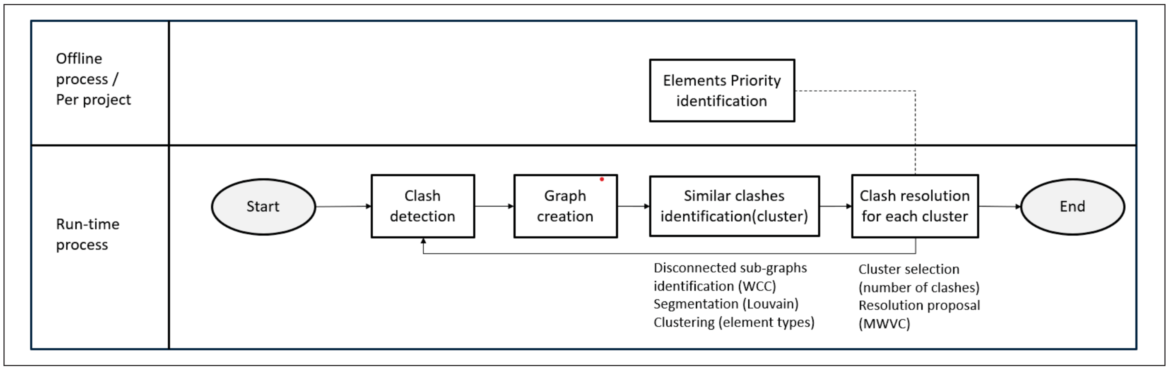


Figure 3.1 High-level process of the proposed artifact.

clash detection takes place and serves as a stable reference throughout the entire coordination process.

The run-time process is executed iteratively for each coordination cycle and consists of four sequential steps: (1) Clash Detection, in which conflicts between building components from different disciplines are identified; (2) Graph Creation, in which detected clashes are transformed into a graph model where components are represented as nodes and clashes as edges; (3) Similar Clashes Identification , in which the graph is analyzed using Weakly Connected Components (WCC) for disconnected subgraph identification, the Louvain algorithm for segmentation, and element type based clustering to group recurring clash patterns; and (4) Clash Resolution for each cluster, in which the Minimum Weighted Vertex Cover (MWVC) algorithm proposes the minimal set of components to modify within each cluster.

This high-level view of the artifact is expanded into the detailed process flow presented in Figure 3.2, which maps each step to its corresponding algorithmic and computational components. The following subsections explain each phase of the artifact in detail.

To achieve these goals, the study uses graph theory, as graph structures clearly show dependencies among building components (Ismail, Nahar & Scherer, 2017). The resulting graph is implemented in a graph database, where building components are represented as nodes and clashes as edges (relationships). This representation supports the analysis of relationships between components and provides a clear view of interconnected clashes, enabling a more structured and organized analysis. The study develops and tests an iterative process.

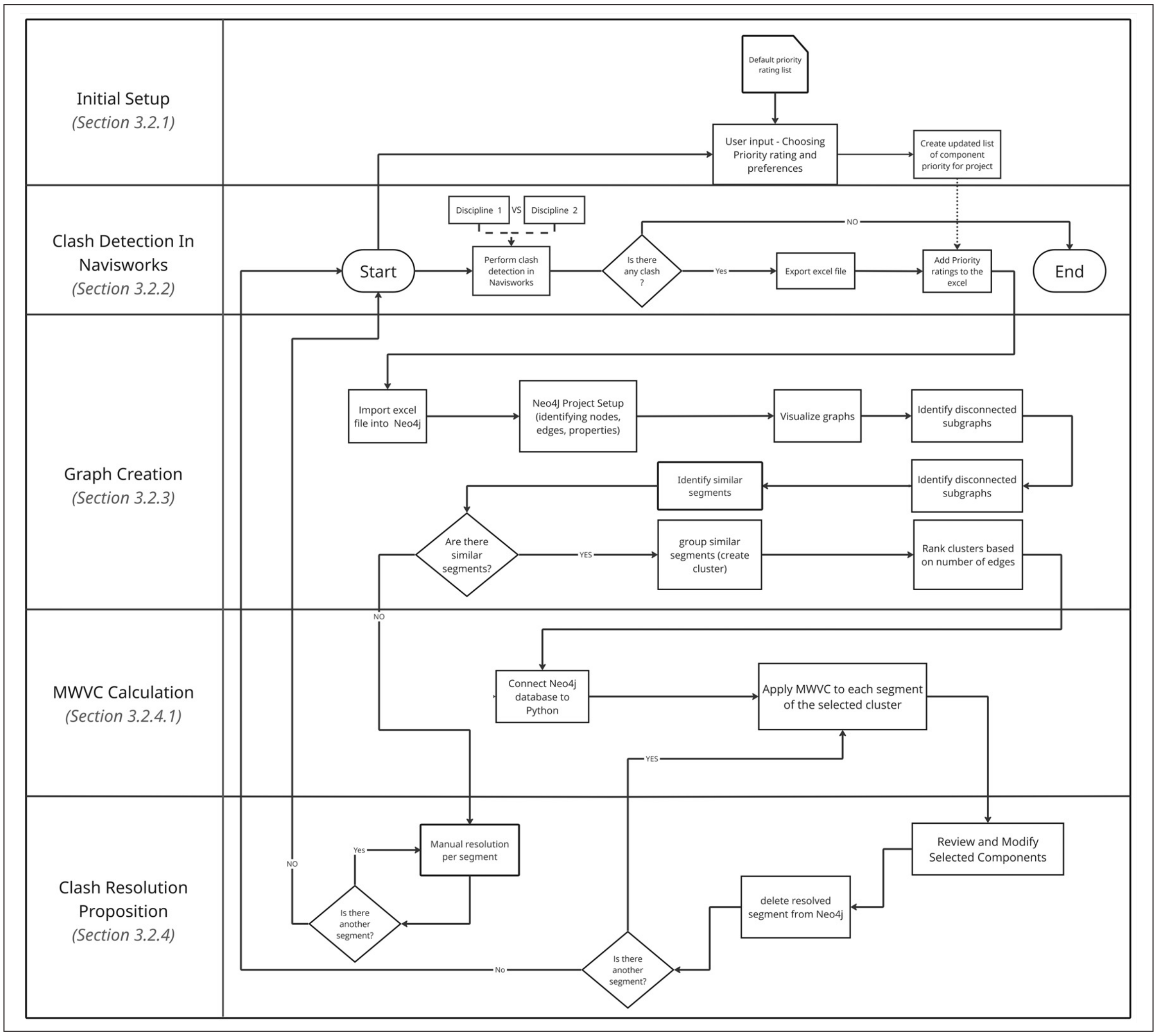


Figure 3.2 Detailed process flow of the proposed artifact.

As illustrated in Figure 3.2, the proposed workflow consists of five main steps: Initial Setup, Clash Detection, Graph Creation, Clash Resolution.

3.2.1 Initial setup

The initial setup of the proposed system established the foundation for the prioritization logic used in the clash-resolution workflow. Throughout this study, priority is defined as a numerical value indicating the ease of modifying a component; components with lower priority values are modified first during clash resolution. This phase ensured that the prioritization logic aligns with real-world coordination practices and reflects the relative flexibility of architectural, structural, and MEP systems during design coordination.

A review of BIM coordination guidelines, system classification frameworks, and modeling standards was first conducted to establish an initial priority structure for building systems. Additionally, a validation step was performed through individual semi-structured interviews with five experienced BIM coordinators actively involved in multidisciplinary projects, including commercial, residential, and industrial developments. The coordinators had between 3 and 10 years of professional experience in BIM-based coordination. Each interview was conducted online, lasted approximately one hour, and followed an open-ended format in which participants were presented with the initial priority list in the form of an Excel file and asked to provide feedback, suggest adjustments, and share their reasoning based on their professional experience. Their responses were analyzed qualitatively, and recurring patterns and consensus were used to refine and adjust the priority rankings. This process ensured that the proposed priority structure reflects practical coordination experience and industry practices.

The interviews revealed a clear consensus regarding the flexibility rating of various building systems during coordination. Architectural and structural systems were consistently identified as the least flexible, as modifications to walls, slabs, columns, or roofs typically lead to geometric, analytical, and downstream impacts. Civil systems including sanitary, stormwater, telecom, and site utilities were also considered limited in flexibility due to slope requirements, fixed connection points, and underground routing constraints.

Large equipment such as HVAC units was ranked slightly lower but remained in the upper portion of the hierarchy due to manufacturer constraints, required clearances, and strict installation tolerances. Gravity-based systems, particularly plumbing networks, were placed in the mid-range of the hierarchy because they must respect elevation and slope requirements, although limited rerouting may be possible within shafts or technical spaces. Mechanical HVAC systems were classified as moderately flexible.

Fire protection systems were considered more flexible, as adjustments typically involve rerouting pipes while maintaining required coverage and spacing. Electrical systems were ranked among the most adaptable disciplines due to their relatively small size and the greater ease of rerouting conduits and cable trays. IT systems were identified as the most flexible components in the hierarchy, as they generally require minimal space and can be relocated with limited impact on other building systems.Overall, the validated priority structure reflects real-world coordination constraints and supports consistent decision-making during clash resolution.

The resulting hierarchy, shown in Figure 3.3, represents the priority structure used in our case study throughout. This pyramid is proposed to serve as the basis for assigning flexibility rating to elements in the graph resulting from clash detection report and directly influences identification of the most suitable component for modification within each clash cluster (more details will be presented in subsection 3.2.3.3). The proposed priority list presented in table can be used as a basis for many projects. However the table can be modified by coordinators based on the specific requirements of each project (see Appendix II-1).

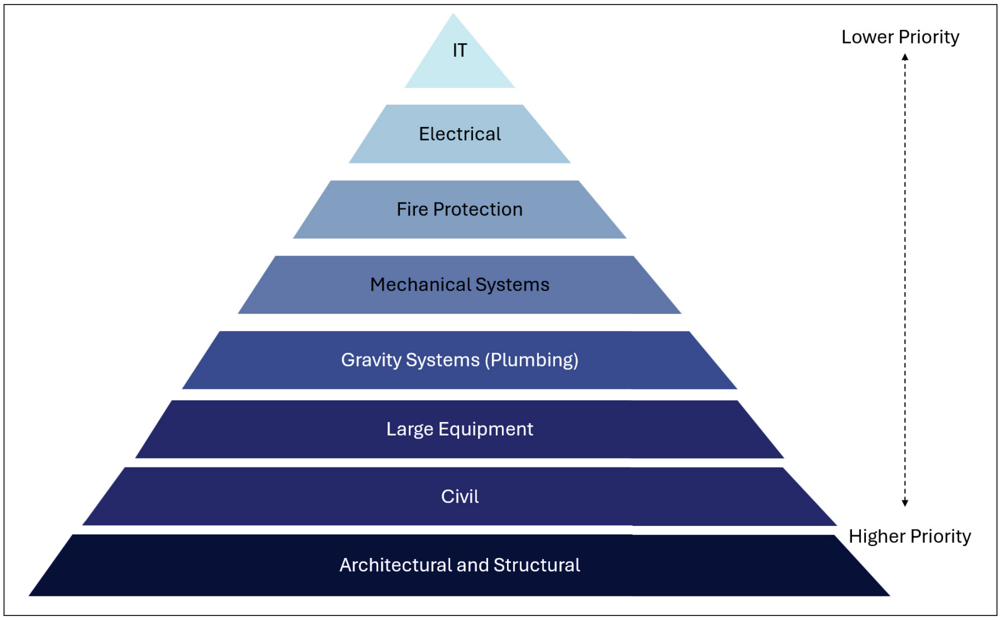


Figure 3.3 System priority structure validated by coordinators.

The creation of the priority table concludes the initial phase of the implementation of the proposed system. It establishes the foundation for the subsequent steps ensuring that all decision-support calculations are informed by industry-approved flexibility rankings. The default list ranks elements from highest to lowest priority.

3.2.2 Clash detection

The second step is to use a clash detection software (e.g., Navisworks) to identify conflicts between two selected disciplines. After completing clash detection, the results are exported as an HTML report and converted it to Excel file for easier analysis. The report provides key details, including Clash Name, Distance, Clash Point, Grid Location, Item IDs, Paths, and Item Names. Next step is to add priority values taken from the priority list to the excel file. This is done by matching the item's system and subsystem information with the corresponding entry in the priority table, and assigning the associated priority value to each item directly in the Excel file. Table 3.1 shows the data structure of the excel table.

Table 3.1 Data structure of the Excel table

Field	**Description**
Item ID	A unique identifier assigned to each building component.
Item Name	The name of the building component.
Path	The location of the element in the model hierarchy.
Priority Rank	The priority value assigned to the component based on its modification importance.
Clash Name	The identifier assigned to the detected clash (e.g., Clash123).
Distance	The penetration distance between the clashing components.
Clash Point	The location where the clash occurs within the model.
Grid Location	The approximate area of the clash based on the building grid system.

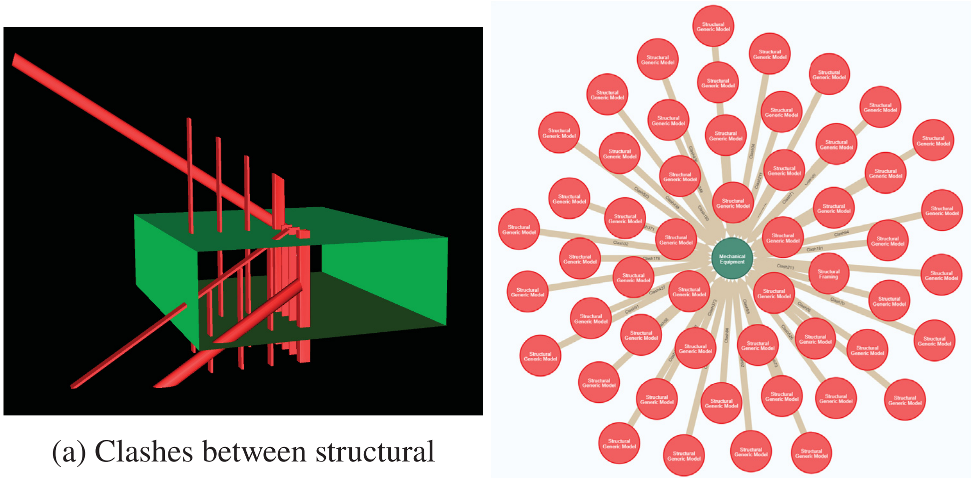
3.2.3 Clash grouping

Due to the interdependence of building systems, multiple clashes may involve the same components; therefore, resolving a single clash can simultaneously address several others (Hu *et al.*, 2019). Grouping related clashes, rather than managing each individually, improves efficiency and supports better decision-making. The objective is to identify similar segments of the graphs, cluster them and apply the same resolution strategy to each cluster.

The third step of the workflow groups related clashes that share common components. The Excel file from the previous stage is imported into a graph database. In this study, the graph database was used to represent and analyze the relationships between building components and their associated clashes. The data were stored and queried using Neo4j and its query language, Cypher, enabling the explicit modeling of components as nodes and clashes as edges. A Neo4j database organizes data through three core elements: nodes, relationships (edges), and properties.

In our proposed approach, each building component involved in a clash is represented as a node. The edge between two nodes represents a clash between those components. Each node and edge includes properties that capture relevant attributes. The properties of each node are Item ID, Item Name, Path, and Priority Rank. The properties of each edge are Clash Name, Distance, Clash Point, and Grid Location.

Two examples illustrating the structure of nodes and edges are shown in figures 3.4 and 3.5.



(a) Clashes between structural generic models and a mechanical equipment shown in Navisworks

(b) Graph-based model of the clash

Figure 3.4 3D BIM model (a) and the graph-based model (b) of a clash.

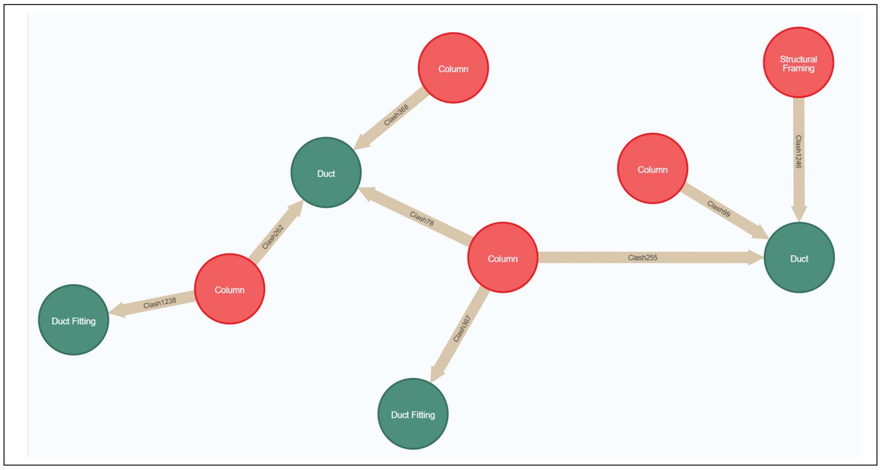


Figure 3.5 Graph-based model of clashes

3.2.3.1 Weakly Connected Components

Neo4j organizes all information within a unified graph structure rather than dividing it into distinct subgraphs. This unified structure provides a complete view of all relationships in the model, which can later be decomposed into smaller parts when required for analysis. To distinguish independent parts of this large graph, the Weakly Connected Components (WCC) algorithm is applied (Newman, 2010). The WCC algorithm was selected because it fits well with how clash data is structured in BIM coordination. In a clash graph, the relationship between two components has no direction; when two components clash, neither one is the cause or the result, it is simply a conflict between them. WCC groups nodes based on whether they are connected, without considering direction, which makes it appropriate for this type of data. In addition, in large BIM projects, not all components interact with each other, so the clash graph naturally splits into separate groups of components that have no conflicts in common. Identifying these independent groups early in the process means that each following step deals with a smaller and more focused set of clashes, making the analysis clearer and more manageable.

The WCC algorithm identifies groups of connected nodes, ignoring relationship direction. Each weakly connected component functions as an isolated island in the network. Nodes within a group connected to each other, but not with nodes in other groups (Bratanic, 2024). Starting graph analysis with the WCC algorithm provides a clear overview of the graph's structure, highlighting its connectivity or fragmentation (Bratanic, 2024). The WCC algorithm is a foundational step in our proposed workflow. Identifying these disconnected subgraphs allows subsequent analyses, such as similarity assessment, to focus on similar groups of clashes.

Figure 3.6 illustrates an example of a clash graph of a project divided into disconnected subgraphs. The clash graph typically contains several disconnected subgraphs. The WCC algorithm can identify these disconnected subgraphs.

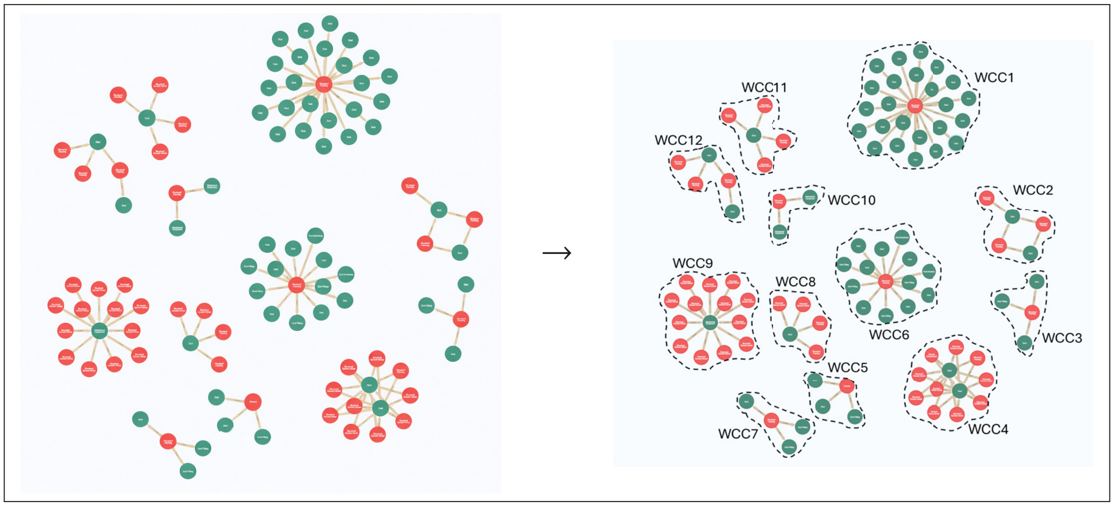


Figure 3.6 Identification of the disconnected subgraphs using weakly connected components (WCC) during the processing workflow.

3.2.3.2 Louvain algorithm

To simplify large and complex clash and support more efficient analysis within each disconnected subgraph, the Louvain algorithm is used to divide the disconnected subgraph into smaller parts called segments.

Blondel, Guillaume, Lambiotte & Lefebvre (2008) introduced the Louvain algorithm, which identifies groups of connected nodes in a graph. The method starts by assigning each node to its own group and then iteratively moves nodes between neighboring groups to improve modularity, as defined in Equation 3.1 (Newman & Girvan, 2004).

$$\Delta Q_{ij} = \frac{1}{2m}\left(d_{ij} - \frac{d_i d_j}{m}\right) \tag{3.1}$$

where m is the total number of edges, d_i and d_j are the degrees of nodes i and j, and d_{ij} is the observed connection between them. The node is reassigned to the neighboring community that yields the highest modularity gain. If no move improves modularity, the node remains in its current community.

After all nodes have been examined, the algorithm completes one iteration. The detected communities are then merged into single units, and the connections between these units are updated to reflect their combined relationships. This simplified structure is used as the input for the next iteration.

The process is repeated until no further improvement in modularity is achieved (Lambiotte, Delvenne & Barahona, 2008). With each iteration, smaller communities are combined into larger and more general ones, forming a hierarchical structure. The algorithm stops automatically when no meaningful improvement is possible. In practice, this occurs after only a limited number of iterations, making the method efficient and suitable for large disconnected graphs (Blondel *et al.*, 2008). After applying the Louvain community detection algorithm, each clash node in the database receives a componentId identifying its disconnected subgraph (from the WCC algorithm) and a subComponentId identifying smaller communities within that subgraph. Throughout this study, the subgraphs identified within each disconnected subgraph by the Louvain algorithm are referred to as segments.

For example, in figure 3.7, the entire subgraph is assigned ID 24 (componentId=24). After applying the Louvain algorithm, two subgraphs were identified: the left part, containing Structural Framing and Mechanical Equipment, subgraph ID is 2 (or subComponentId=2) (24_2), and the right part, containing Floor and Duct Insulation elements, subgraph ID is 4 (or subComponentId=4) (24_4). This componentId_SubcomponentID naming convention was used for consistent labeling. It enables quick tracking of each group's position within the graph and facilitates comparison of similar clash patterns across the project.

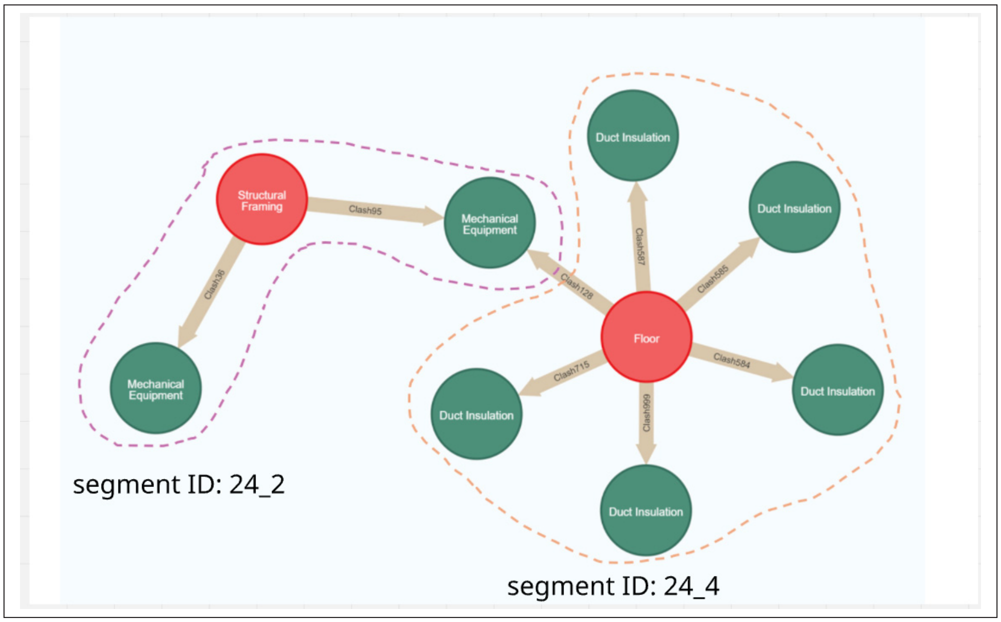


Figure 3.7 Graph-based representation of clash segments (segment IDs (24_2) and (24_4)).

The Louvain algorithm was selected to detect communities within each disconnected subgraph because it suits the repeated nature of BIM coordination. One of its key advantages is that, unlike other community detection methods that require the number of groups to be set in advance, it finds the best grouping automatically, which is useful in clash management since the number of meaningful clash groups is not known before the analysis. It is also more computationally efficient than other available methods, meaning it can be run repeatedly across multiple coordination cycles without slowing down the process.

3.2.3.3 Finding clusters of similar segments

This section explains the proposed method identify similar segments (identified by the Louvain algorithm) that involve the same pair of building element types. The goal is to find recurring clash patterns. Similar sets of segments are grouped and referred to as a cluster, representing repeated clash patterns for which a consistent resolution strategy can be applied.

For each segment, the component types involved in the clashes are identified. These component types are merged and filtered to retain only clusters that involve exactly two distinct component types, representing direct interactions between two building systems. All segments sharing the same pair of component types are subsequently grouped together, forming clusters of similar clash segments that occur in different locations of the project but involve the same system combination.

The number of clash relationships (edges) within each cluster is then calculated to rank the clusters from the highest to the lowest number of clashes.

To conclude, the grouping phase restructures clash data into a hierarchical graph structure. Building components and clashes are modeled as nodes and edges, disconnected subgraphs are identified using WCC, and segments (Subgraphs within each disconnected subgraph) are detected through the Louvain algorithm. Segments involving the same element-type pairs are then clustered and ranked based on the number of edges (clashes) in each cluster. This process organizes the dataset into analyzable groups and establishes a basis for the subsequent resolution step.

3.2.4 Clash resolution

The third phase of the workflow focuses on the proposed assisted clash resolution. In practical terms, the objective is to determine the smallest set of building elements that should be modified so that clashes within a selected segment are addressed while minimizing overall impact of modification. Instead of resolving clashes individually, this approach identifies the components whose adjustment can eliminate multiple related conflicts simultaneously. To implement this logic computationally, the graph data is processed using a Python implementation of the Minimum Weighted Vertex Cover (MWVC) algorithm, integrated with the Neo4j database.

A vertex cover (VC) in a graph is defined as a set of nodes such that every edge is incident to at least one vertex in the set, ensuring that all edges are collectively covered. In the context of clash management, each node represents a building component, and each edge represents a clash; therefore, selecting a node equates to choosing a component whose modification resolves all associated clashes. The Minimum Vertex Cover (MVC) problem extends the vertex cover definition by seeking the smallest possible vertex cover, or the minimum number of components required to address all clashes within a segment.

Because building components differ in flexibility and modification cost, the MWVC extends the MVC by assigning each component a weight based on its priority or ease of modification. The objective is to identify a set of components that minimises the total weight of the selected elements, rather than simply the number of selected components. Formally, MWVC seeks to minimise the sum of the weights $\sum_{v \in S} w(v)$, where S represents the set of selected components and $w(v)$ is the weight assigned to component v. This optimisation is subject to the condition that for every clash $(u, v) \in E$, at least one of the two components involved in the clash must be included in the selected set S. In this formulation, E denotes the set of clashes (edges) between components, while u and v represent the two components connected by a clash. In this context, $w(v)$ represents the modification priority assigned during the methodology's initial setup. MWVC is an NP-hard optimisation problem (Balaji, Swaminathan & Kannan, 2010), indicating that no polynomial-time exact solution exists for large graphs. However, heuristic and approximation algorithms can yield near-optimal solutions well suited to real-world BIM coordination.

In this study, the MWVC algorithm is applied to each segment to identify the optimal set of building components for modification. By identifying the building components that require adjustment, this method transforms a complex coordination challenge into a structured optimisation process. Modifying only these components resolves all associated clashes within the network, ensuring an efficient solution that aligns with practical engineering constraints.

3.2.4.1 MWVC algorithm

The MWVC is a typical NP-hard problem (Karp, 2009), and various algorithms have been developed to address it. These approaches fall into three main categories: exact, approximation, and local search methods (Hochba, 1997).

Exact algorithms, including Integer Linear Programming (ILP) and branch-and-bound, seek the optimal solution by exhaustively searching the solution space or using mathematical formulations (Garey, Johnson *et al.*, 1990). Approximation methods, such as greedy or linear programming (LP)-based techniques, aim for near-optimal solutions within polynomial time by applying heuristics or relaxation strategies (Bar-Yehuda & Even, 1981). Local search or metaheuristic methods, such as simulated annealing, tabu search, and genetic algorithms, explore neighboring solutions and are often used for large-scale or complex problem instances.

For this study, which focuses on providing assistance for clash resolution within BIM coordination models, an approximation approach was selected. Although the graph is first decomposed into smaller segments, each segment may still contain a dense set of interconnected components. Applying exact algorithms to each component remains computationally demanding and impractical for iterative coordination workflows. Metaheuristic methods, while capable of producing high-quality solutions, require extensive parameter tuning and longer convergence times. In contrast, a greedy approximation approach provides an effective balance between computational efficiency and solution quality, making it well suited for repeated execution across multiple segments and for supporting timely decision-making in clash management.

A greedy algorithm iteratively makes locally optimal choices, aiming for a near-optimal global solution (Hochba, 1997). While it does not guarantee an exact minimum vertex cover, it delivers fast and scalable results suitable for time-sensitive applications. For MWVC, the algorithm begins with all edges uncovered and repeatedly selects the vertex with the highest degree-to-weight ratio (Bar-Yehuda & Even, 1981). After each selection, the covered edges are removed from the graph, and the process continues until all edges are covered. After each selection, all incident edges are removed, and the process repeats until the graph is fully covered.

This iterative selection process is similar to the maximum coverage problem, where subsets are chosen based on the number of new elements they cover at each step (Hochba, 1997). The greedy MWVC heuristic aims to minimize total weight while maximizing coverage efficiency. Its simplicity enables polynomial-time implementation and easy integration with existing graph-based workflows. In practice, it performs well in dense graphs, where a small set of high-impact nodes can cover many edges. Algorithm 3.1 shows the pseudocode for the greedy MWVC calculation.

The figure 3.8 summarizes the main computational steps used to transform clash data into a graph, identify related clash groups, and select components for modification.

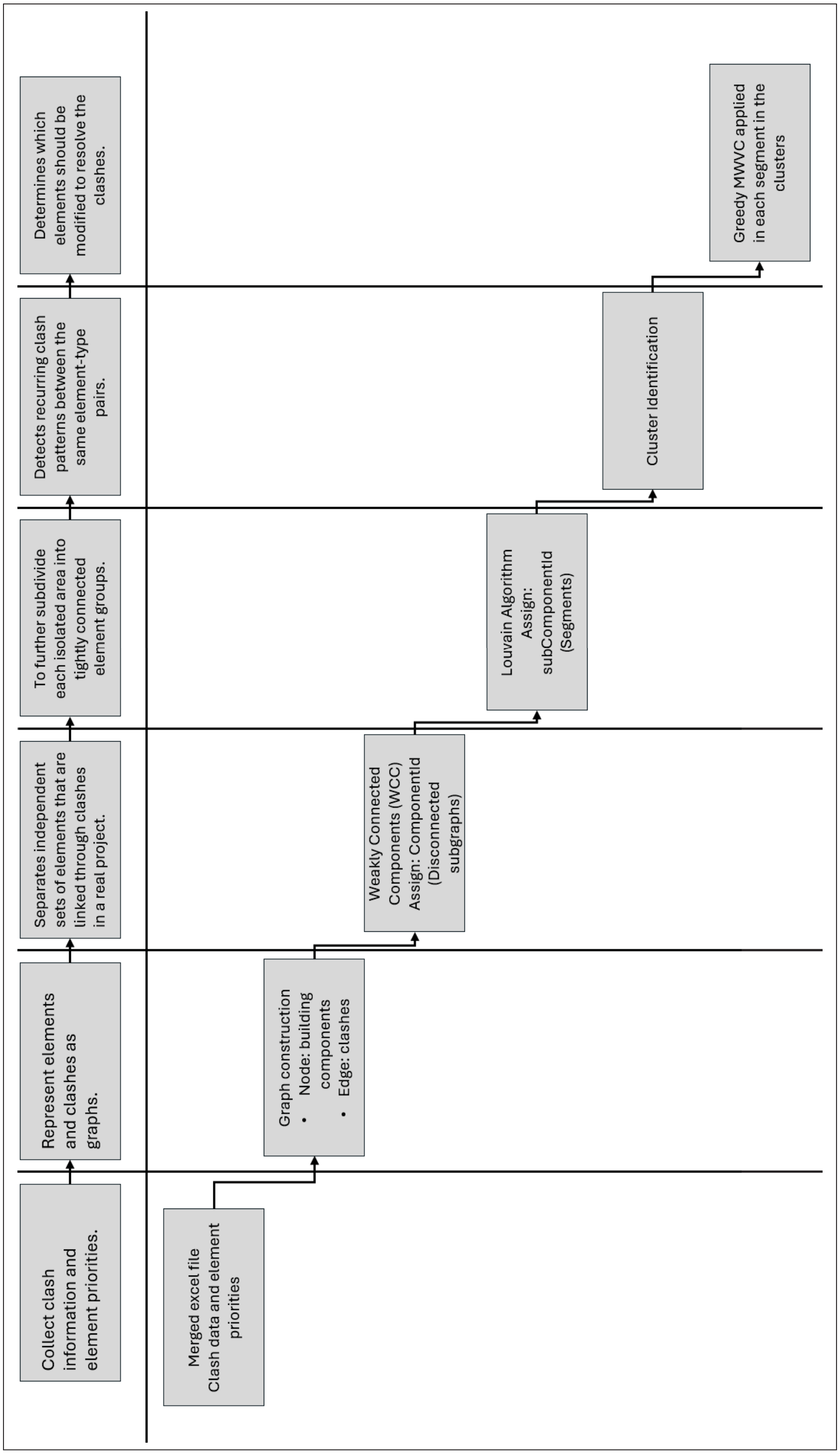


Figure 3.8 Code-level process flow for the proposed system.

Algorithm 3.1 Pseudocode for calculating MWVC

```
Input: Set of target groups $\mathcal{G} = \{g_1, g_2, ..., g_n\}$, database connection parameters
       $(URI, AUTH)$
Output: Result list $\mathcal{R}$ containing node count, edge count, cover set, and total weight per
        group

Initialize $\mathcal{R} \leftarrow [\ ]$;
Open database connection with $(URI, AUTH)$;
Open session $s$;
for each group $g_i \in \mathcal{G}$ do
    $rows \leftarrow$ FetchGroupRows$(s, g_i)$;
    if $rows$ is empty then
        Append $(g_i, 0, 0, [\ ], 0.0)$ to $\mathcal{R}$;
        continue;
    end if
    $G \leftarrow$ BuildGraph$(rows)$;
    $(cover, total\_w) \leftarrow$ MWVC_Greedy_Ratio$(G,$ "weight"$)$;
    Append $(g_i,\ |V(G)|,\ |E(G)|,\ \text{sorted}(cover),\ total\_w)$ to $\mathcal{R}$;
end for
Close session and database connection;
return $\mathcal{R}$;

Subfunction FetchGroupRows$(session, gid)$;
    Run query GET_GROUP_EDGES with parameter $gid$;
    return list of rows from query result;

Subfunction BuildGraph$(rows)$;
    Initialize empty graph $G$;
    for each record $r \in rows$ do
        $u \leftarrow r[u],\ v \leftarrow r[v]$;
        $w_u \leftarrow float(r[wu]),\ w_v \leftarrow float(r[wv])$;
        Add node $u$ to $G$ with weight $w_u$;
        Add node $v$ to $G$ with weight $w_v$;
        Add edge $(u, v)$ to $G$;
    end for
    return $G$;
```

3.3 System design

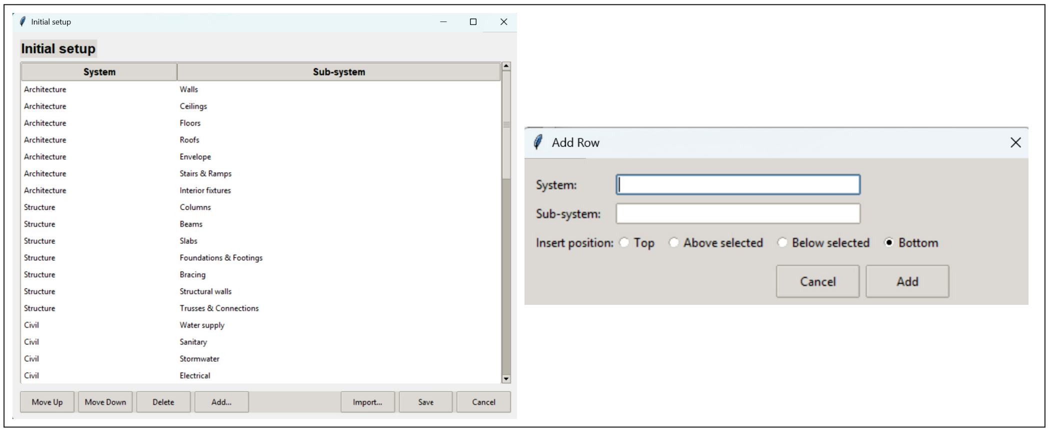
This section describes the main functionalities of the system and explains how it is intended to interact with the user during the clash management process. The proposed artifact was developed using two main technologies. The MWVC algorithm and the overall workflow logic were implemented in Python, while the graph database queries and community detection algorithms were written in Cypher, the native query language of Neo4j (version 2025.12.1). The graphical user interface (GUI) was developed using Tkinter, a standard Python library for desktop application development.

The clash detection is performed within Navisworks, and the resulting clash report is exported as an HTML file. This file is then manually converted into an Excel spreadsheet, which serves as the input to the artifact. Priority values are added to the Excel file by matching each element's system and subsystem information with the corresponding entry in the priority table. The enriched Excel file is subsequently imported into the Neo4j graph database, where the Python script connects to Neo4j to execute the graph construction, clustering, and MWVC calculations. The output of the artifact is generated as an Excel/CSV report summarizing the components to be modified and their assigned disciplines, which can then be used by BIM coordinators to guide the resolution process.

The information exchanged between tools therefore relies on three main formats: HTML for the clash report exported from Navisworks, Excel/CSV for both the input file and the final resolution report, and Cypher queries for communication between the Python implementation and the Neo4j database.

The codebase of the proposed artifact is not currently published in a public repository. However, it is available upon request from the author.

As delineated in sub-section 3.2.1, in the Initial Setup phase, the user begins by reviewing and customizing the default element priority list. This list serves as the foundation for the entire workflow, defining the priority values of each building component. The user can modify or reorder this list to reflect the coordination logic of the specific project for example, assigning higher priority values to structural components that are less flexible to modify, while giving lower priority to mechanical or electrical systems that can be more easily adjusted. Once this setup is completed, the system saves the customized list as an Excel file and uses it as a reference throughout the following phases figure 3.9 shows the GUI for initial setup.



System	Sub-system
Architecture	Walls
Architecture	Ceilings
Architecture	Floors
Architecture	Roofs
Architecture	Envelope
Architecture	Stairs & Ramps
Architecture	Interior fixtures
Structure	Columns
Structure	Beams
Structure	Slabs
Structure	Foundations & Footings
Structure	Bracing
Structure	Structural walls
Structure	Trusses & Connections
Civil	Water supply
Civil	Sanitary
Civil	Stormwater
Civil	Electrical

Insert position: ○ Top ○ Above selected ○ Below selected ● Bottom

Figure 3.9 GUI for Initial Setup

The second phase, Clash Detection, is carried out within a BIM coordination platform such as Navisworks. Here, the user performs clash detection to identify geometric conflicts between elements of different disciplines. After the detection is completed, the system verifies whether any clashes exist. If no clashes are detected, the workflow ends. However, if clashes are present, the user proceeds to the next phase for deeper analysis and grouping.

During the Clash Grouping phase, the user initiates the process of transforming the detected clash data into a graph representation, while the priority list stored in the Excel file is also imported into the graph database (Neo4j) and associated with the corresponding building elements. In this graph, building elements are modeled as vertices and clashes as edges connecting those nodes as described in sub-section 3.2.3. The user then instructs the system to create disconnected subgraphs using WCC (sub-section 3.2.3.1), then chopping these disconnected subgraph into smaller segments using Louvain algorithm (sub-section 3.2.3.2). The system automatically groups similar segments based on the element type, which helps the user focus on specific categories of clashes. The user can then select and visualize the highest priority cluster (the group of clashes that contains highest number of clashes) directly in Navisworks to better understand its spatial context.

In the final phase, Clash Resolution, the user executes the MWVC algorithm on selected segment. This algorithm identifies the smallest number of elements that need to be modified to eliminate all clashes in the selected segment, while minimizing the total modification cost according to the defined priorities. The system then visualizes the selected elements in Navisworks, allowing the user to confirm the proposed solution. A detailed report is automatically generated using Python language, summarizing the components to be modified and assigning them to the corresponding discipline. Once the resolution is approved and completed, the user can delete the resolved cluster in Neo4j from the active list to keep the clash database updated. Figure 3.10 shows the GUI for clash grouping and MWVC calculation.

Types	Similar Segments	Total Edges	MWVC	Delete
[Structural Equipmen	[42_2, 69_2, 69_5, 70_8, 1	90	Run MWVC	Delete
[Floor, Duct]	[69_1, 70_1]	72	Run MWVC	Delete
[Structural Framing, D	[16_4, 16_3, 33_5, 33_3, 6_	18	Run MWVC	Delete
[Structural Framing, D	[63_4, 33_6, 320_1, 25_50,	12	Run MWVC	Delete
[Structural Framing, M	[24_2, 63_1, 147_1, 179_1,	8	Run MWVC	Delete
[Structural Framing, A	[157_10, 162_5, 157_15, 1	7	Run MWVC	Delete
[Structural Framing, D	[292_1, 842_1, 1087_1, 10	5	Run MWVC	Delete
[Column, Duct]	[19_2, 274_1, 465_1]	4	Run MWVC	Delete
[Structural Equipmen	[42_6, 551_1]	4	Run MWVC	Delete
[Structural Framing, D	[305_1, 0_67]	3	Run MWVC	Delete

Segment ID	Number of Components	Number of Clashes	Cover	Element ID
178_2	21	20	1	1366541
263_10	17	16	1	1331720
277_7	15	14	1	1331718
283_6	12	11	1	1368870
422_15	21	20	1	1331719
406_31	41	40	1	1366367
406_32	41	40	1	1366368
452_6	15	14	1	1366371
406_33	41	40	1	1366369
406_34	41	40	1	1366370
491_1	13	12	1	1331732
492_3	19	18	1	1331727
499_2	9	8	1	1366229
562_2	8	7	1	1340673
685_20	27	26	1	1366542
902_5	6	5	1	1361681
932_10	12	11	1	1361678
941_2	4	3	1	1361680

Figure 3.10 User steps

CHAPTER 4

RESEARCH RESULTS

This chapter reports the main findings of the research project obtained through the evaluations and data analysis. Two case studies selected to evaluate the research constructs proposed in Chapter 3. The results for each case study are presented in the following subsections. Table 4.1 provides basic information about these projects.

Table 4.1 Case study projects information summary

Project Name	**Location**	**Project Function**	**Gross Floor Area**
Hospital Complex	Québec City, Canada	Health and Research Facility	78,100 m^2
Office Building	Lévis, Canada	Commercial Office	1,550 m^2

4.1 Findings from case study 01:A Hospital

4.1.1 Project Presentation

The selected project, commissioned by the Société Québécoise des Infrastructures (SQI), is part of the healthcare sector and was designed to expand the Enfant-Jésus Hospital complex. The delivery mode followed a construction management approach, enabling continuous coordination among architectural, structural, and MEP teams. Phase 1 of the project, covering approximately 78,100 m^2, was completed in 2020 and included several major components. It involved the development of three temporary parking sites to accommodate construction logistics, the construction of the Integrated Cancer Centre comprising radiation oncology units, biochemistry laboratories, backup generators, and technical tunnels connecting multiple buildings, as well as the creation of a multi-level underground and above-ground parking lot totaling 36,000 m^2 with a capacity of around 800 parking spaces. The project also included site landscaping across 44,000 m^2 to improve pedestrian and vehicular circulation, the demolition of the Partagec building (15,000 m^2) to prepare for new facilities, and refurbishment work within existing hospital areas covering 500 m^2.

4.1.2 Implementation of the Proposed Methodology (Structural vs Ventilation)

This section presents the implementation of the proposed process flow on a real coordination case involving structural and ventilation systems. The implementation was carried out by the researcher provided by the project team and integrated within a BIM coordination environment.

4.1.2.1 Initial Step

For this case study, we used the default priority list from the methodology and made no changes (table 3.1).

4.1.2.2 Clash detection

In this project, clash detection was provided by the construction company responsible for the project. The company manually performed clash detection and grouping in Navisworks Manage, producing well-structured examples. The dataset shared with us included these manually grouped clashes. This provided a useful basis to compare our graph-based grouping and resolution method with the company's manual approach.

We selected the clash dataset between the Structural and Ventilation systems. This set serves as a benchmark for evaluating our automated grouping method against company's manual results. The clash detection process identified 1255 conflicts between structural and ventilation elements.

4.1.2.3 Graph Creation

After importing the clash data into Neo4j, a graph-based data model was created in which each node represents a building component and each edge represents a clash relationship between two components. Attributes such as component type, discipline, and priority were assigned to each node, while edge properties included the clash identification number and the corresponding clash name. The graph construction process resulted in a total of 1,235 nodes and 1,255 edges, representing all components and their respective clash relationships identified in the case study. as illustrated in figure 4.1.

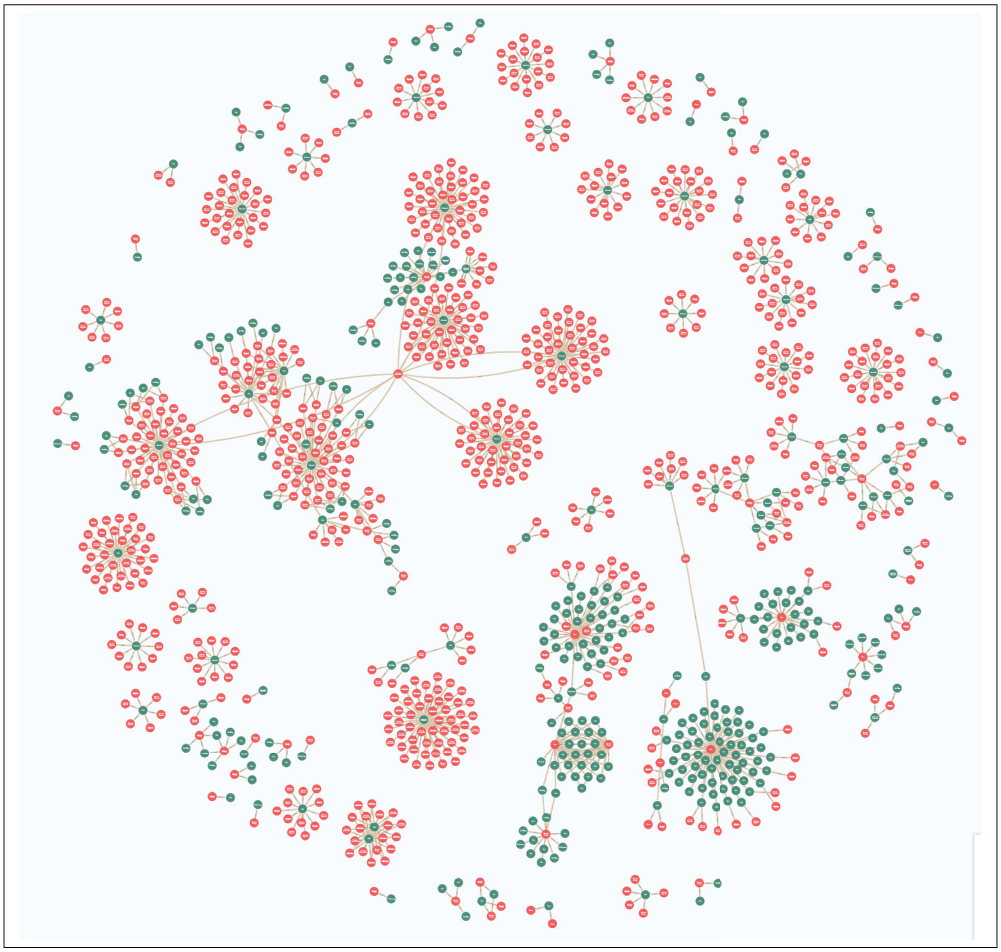


Figure 4.1 Graph representation of the Hospital case study (Structural vs Ventilation).

Once the graph was built in Neo4j, the WCC algorithm helped find isolated subgraphs in the network. This process revealed 91 disconnected subgraphs, each made up of elements linked by clashes and separate from other groups in the model. Next, the Louvain community detection algorithm divided these subgraphs into segments. The Louvain method produced 151 segments, each representing a local area with many clashes. A similarity analysis then grouped and merged segments with similar element types, forming sets of segments with shared characteristics. Table

4.2 shows the results of this clustering, listing the final clusters from the most to the fewest clashes.

A total of 12 clusters were generated. Table 4.2 presents the results of the similar segment identification process, which groups together subgraphs that involve the same types of component interactions. Each row lists a pair of element types (for example, Structural Equipment and Air Terminal), followed by the group IDs of the subgraphs that share similar characteristics. The last column shows the total number of edges, which represents the total number of clash relationships between those two component types.

In other words, the table summarizes how many clashes exist between specific system pairs and which segment IDs contain those relationships. For example, the first row shows that multiple subgraphs, such as 178_2, 263_10, 277_7, and others, involve clashes between Structural Equipment and Air Terminal elements, with a total of 345 edges detected. This means that across all those subgraphs, 345 clash connections were found between these two types of elements.This grouping approach helps identify repetitive clash patterns between certain systems and supports further clustering to simplify large and complex coordination datasets. Based on the results, it is recommended that the user begins the resolution process with the cluster containing the highest number of edges.

Table 4.2 Similar segments and total edges between Structural and Ventilation

Cluster	**Types**	**Similar Segments**	**Similar Segments Number**	**Total Edges**
1	[Structural Equipment, Air terminal]	[178_2, 263_10, 277_7, 283_6, 422_15, 406_31, 406_32, 452_6, 406_33, 406_34, 491_1, 492_3, 499_2, 562_2, 685_20, 902_5, 932_10, 941_2]	18	345
2	[Structural Equipment, Duct]	[42_2, 69_2, 69_5, 70_8, 151_1, 70_10, 70_14, 69_11, 0_39, 0_40, 0_41, 0_44, 0_50, 0_54, 0_56, 318_1, 321_5, 0_59, 69_23, 69_24, 69_26, 69_27, 69_29, 69_33, 69_34, 69_35, 0_64, 418_3, 419_1, 438_1, 464_1, 798_7, 805_1, 944_6, 1133_12]	35	90
3	[Floor, Duct]	[69_1, 70_1]	2	72
4	[Structural Framing, Duct]	[16_4, 16_3, 33_5, 33_3, 60_1, 255_2, 262_1, 264_1, 270_1, 441_1, 580_1, 589_1, 1040_1, 1042_1]	14	18

Table 4.2 Similar segments and total edges between Structural and Ventilation (continued)

Cluster	**Types**	**Similar Segments**	**No. of Similar Segments**	**Total Edges**
5	[Structural Framing, Duct Fitting]	[63_4, 33_6, 320_1, 25_50, 571_1, 726_1]	6	12
6	[Structural Framing, Mechanical Equipment]	[24_2, 63_1, 147_1, 179_1, 988_1]	5	8
7	[Structural Framing, Air terminal]	[157_10, 162_5, 157_15, 1039_1]	4	7
8	[Structural Framing, Duct Accessory]	[292_1, 842_1, 1087_1, 1088_1, 1195_1]	5	5
9	[Column, Duct]	[19_2, 274_1, 465_1]	3	4
10	[Structural Equipment, Duct Fitting]	[42_6, 551_1]	2	4
11	[Structural Framing, Duct Insulation]	[305_1, 0_67]	2	3
12	[Column, Duct Fitting]	[43_6, 176_1]	2	2

4.1.3 Comparison between manual and proposed process flow

This section compares the manual clash grouping performed in Navisworks with the proposed system. The objective is to evaluate how well the proposed flow reproduces the logic applied by the BIM coordinator, while also assessing its efficiency.

Both approaches analyzed the same set of 1,255 clashes. The manual workflow grouped these clashes into 55 groups, while the proposed system identified 91 disconnected subgraphs using the WCC algorithm, which were further subdivided into Louvain segments and organized into clusters of similar clash patterns. Although the proposed workflow produced more initial groups, they were more detailed and systematically organized based on component-type relationships.

4.1.3.1 MWVC Calculation

After the clustering phase, the MWVC algorithm was applied to each group to identify the optimal set of components to modify for efficient clash resolution. The MWVC algorithm aims to find the smallest set of nodes that collectively cover all edges (clashes) in a cluster, while minimizing the total modification cost based on element priority weights.

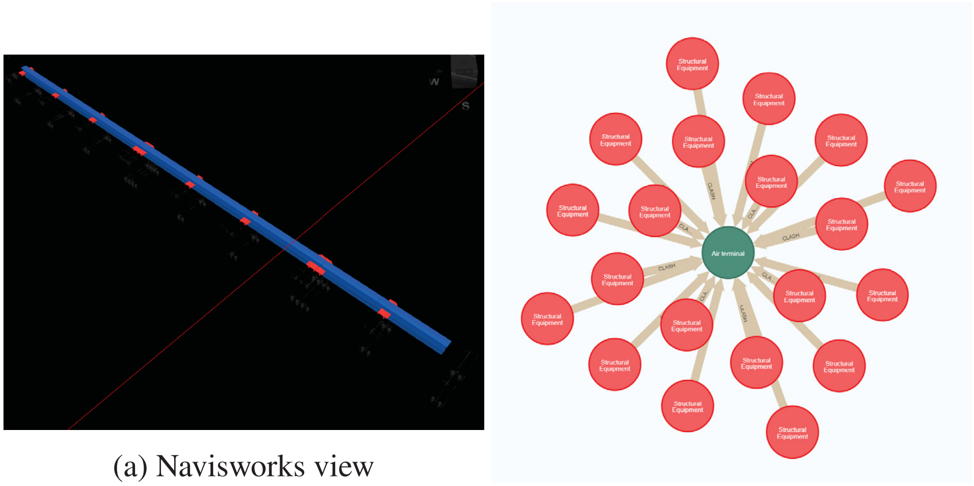
In all segments, the MWVC algorithm identified a cover size of 1, indicating that a single node in each subgraph could resolve all related clashes. This result highlights the algorithm's ability to automatically pinpoint the most influential component in each localized clash network. The selected element IDs (e.g., 1366541, 1366367, 1368870) represent higher flexible elements for modification.

Table 4.3 presents the detailed results for the [Structural Equipment – Air Terminal] cluster, including each detected segment's ID, number of nodes and edges, minimum vertex cover, and the representative element ID selected as the optimal clash resolution candidate.

Table 4.3 Detailed results for
Structural Equipment–Air Terminal subgraphs

Segment ID	**Nodes**	**Edges**	**Cover**	**Element ID**
178_2	21	20	1	1366541
263_10	17	16	1	1331720
277_7	15	14	1	1331718
283_6	12	11	1	1368870
422_15	21	20	1	1331719
406_31	41	40	1	1366367
406_32	41	40	1	1366368
452_6	15	14	1	1366371
406_33	41	40	1	1366369
406_34	41	40	1	1366370
491_1	13	12	1	1331732
492_3	19	18	1	1331727
499_2	9	8	1	1366229
562_2	8	7	1	1340673
685_20	27	26	1	1366542
902_5	6	5	1	1361681
932_10	12	11	1	1361678
941_2	4	3	1	1361680

For example figure 4.2 presents the segment (178_2) corresponding to the [Structural Equipment–Air Terminal] cluster. The visualization in Navisworks (left) highlights the spatial location of the clashes, where the blue element represents the component identified for modification according to the MWVC result. The Neo4j representation (right) shows the same segment as a graph structure, where one air terminal node is connected to several structural equipment nodes, confirming that this clash group is centered around a single air terminal component.



(a) Navisworks view

(b) Graph representation

Figure 4.2 Visual comparison of segment (178_2) in Navisworks (a) and its corresponding graph representation (b).

Figure 4.3 presents the [Structural Equipment–Air Terminal] cluster. The elements highlighted in blue represent the components identified by the MWVC algorithm as the ones requiring modification. These elements correspond to the results summarized in Table 4.5. The remaining clusters listed in Table 4.2 were analyzed using the same procedure to identify the elements to be modified and to support clash resolution.

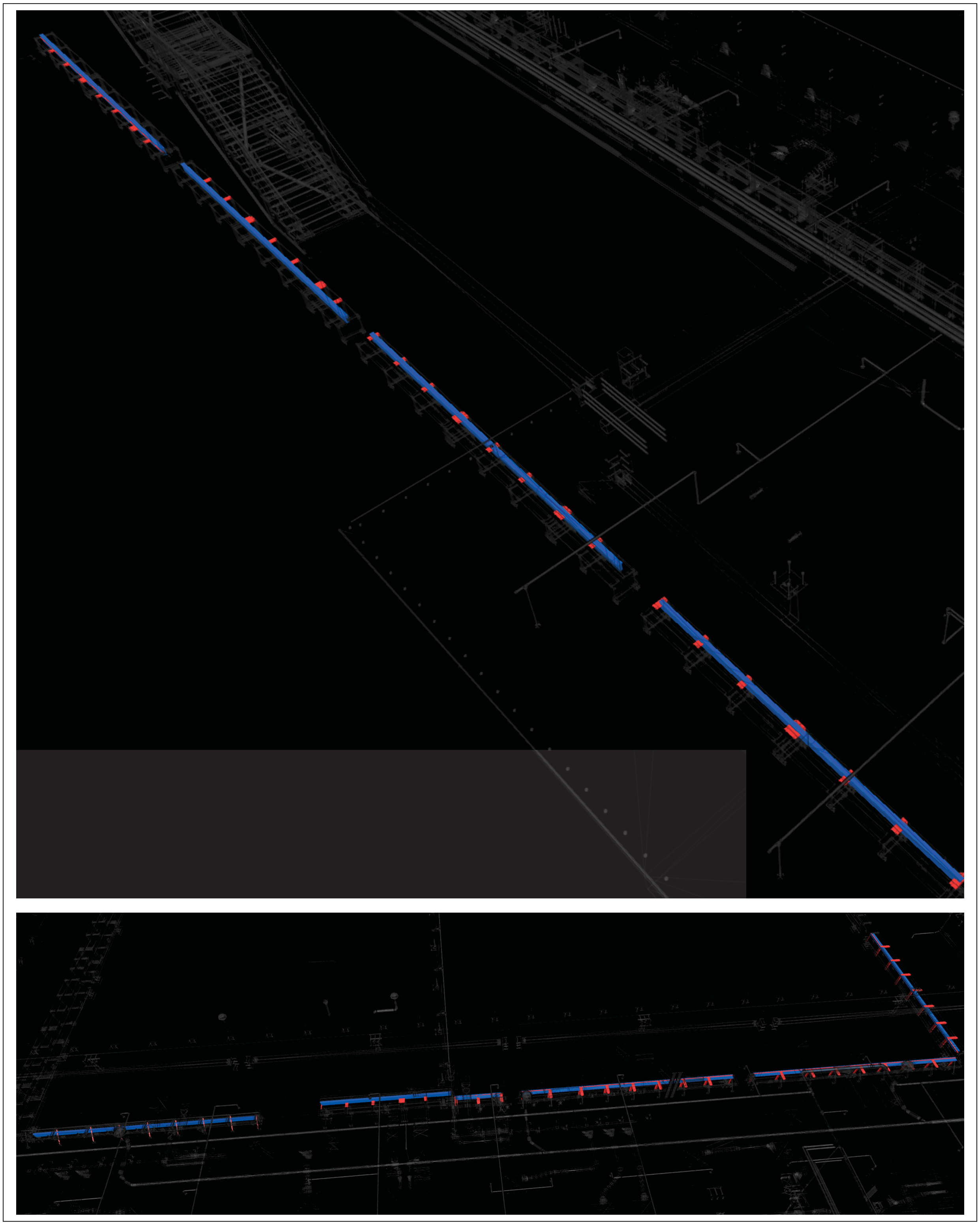


Figure 4.3 Detailed visualization of the [Structural Equipment–Air Terminal] cluster in the Navisworks environment.

4.1.4 Implementation of the proposed workflow (Piping vs Ventilation)

4.1.4.1 Initial Step

For the second run of hospital case study, the clash dataset between the Piping and Ventilation systems was analyzed. As in the previous example, the default priority list from the methodology was used without modification.

4.1.4.2 Clash Detection

This dataset was extracted from the same hospital project and contains all clashes involving piping elements and ventilation components. The clash detection process identified a total of 174 clashes between Piping and Ventilation systems. These clashes were manually exported from Navisworks and served as the input for the automated workflow in Neo4j.

4.1.4.3 Graph Creation

The clash data was imported into Neo4j to construct the graph model. As before, each component involved in a clash was represented as a node, and each clash formed an edge between two nodes. The constructed graph contains 174 edges, each representing a clash between piping and ventilation components. Figure 4.4 illustrates the complete graph.

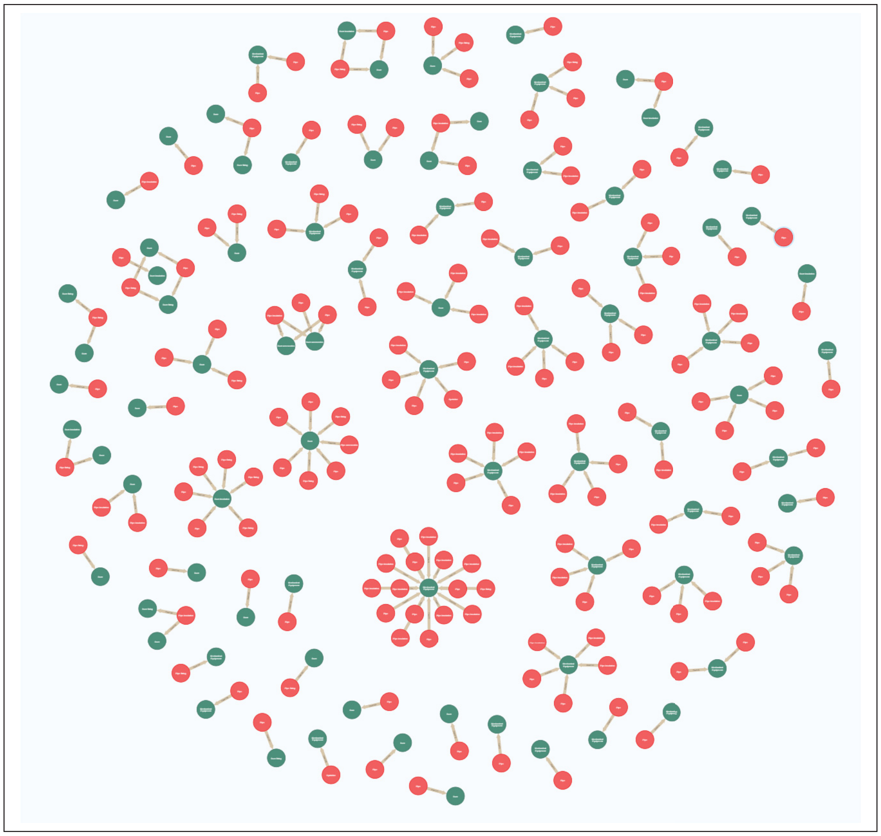


Figure 4.4 Graph representation of the Hospital case (Piping vs Ventilation).

The WCC algorithm was applied first to identify disconnected subgraphs within the network. These subgraphs represent localized clusters of pipe and ventilation elements that clash with each other but not with other areas of the model. The Louvain community detection method was then applied to each subgraph.

Once the Louvain segments were produced, a similarity analysis was performed to group segments based on shared combinations of element types. This process generated five clusters for the Piping–Ventilation case, each representing a specific type of interaction between the two trades.

Table 4.4 summarizes these clusters, listing the relevant segment IDs and the total number of edges (clashes) associated with each element-type pair.

The largest cluster consisted of interactions between pipe and mechanical equipment, with 20 segments and 28 total edges. Other important system interactions included Pipe–Duct (14 edges), Pipe insulation–Duct (7 edges), and smallest clusters involving Duct insulation and Pipe fittings.

Table 4.4 Similar segments and total edges between component type pairs

Types	**Similar Segments**	**Total Edges**
[Pipe, Mechanical Equipment]	[2_1, 9_1, 12_1, 13_1, 18_2, 21_3, 25_2, 28_1, 35_2, 37_1, 42_1, 45_1, 51_1, 63_1, 74_1, 78_1, 84_2, 89_1, 205_1, 206_1]	28
[Pipe, Duct]	[30_4, 97_1, 122_1, 69_1, 185_1, 198_1, 204_1, 212_1, 214_1, 216_1, 217_1]	14
[Pipe insulation, Duct]	[49_1, 69_3, 106_1, 193_2]	7
[Pipe, Duct insulation]	[40_1, 113_1]	2
[Pipe fitting, Duct]	[192_1, 208_1]	2

4.1.4.4 MWVC calculation

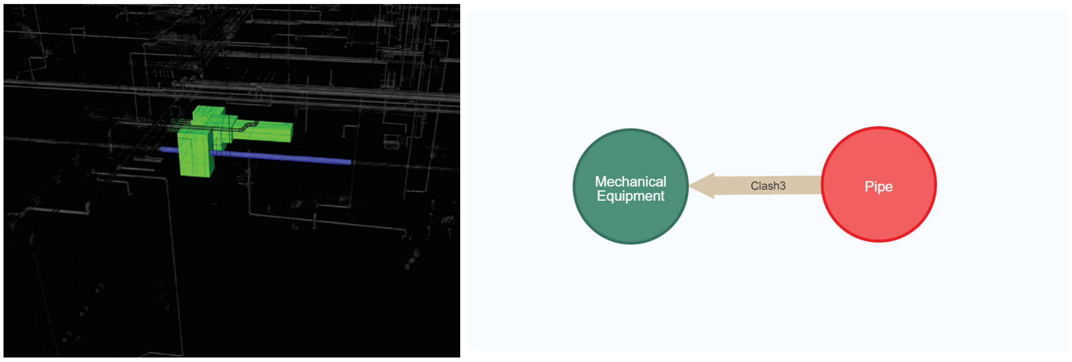
After clustering, the MWVC algorithm was applied to each segment within the largest cluster, Pipe–Mechanical Equipment. This analysis aimed to identify the set of most influential component in each localized clash group.

Across the 20 segments, the MWVC algorithm again selected one element per subgraph as the optimal candidate for modification.

Table 4.5 presents the detailed results, including the number of nodes and edges in each segment and the element ID chosen as the key resolution component.

Table 4.5 MWVC results for [Pipe, Mechanical Equipment]

Segment ID	Nodes	Edges	Cover	Element ID
2_1	2	1	1	4429728
9_1	2	1	1	4627920
12_1	2	1	1	5165772
13_1	2	1	1	8159264
18_2	3	2	2	5639584, 4462802
21_3	4	3	3	6881239, 5640848, 6881284
25_2	3	2	2	4419660, 4419637
28_1	2	1	1	4563011
35_2	4	3	3	4582422, 4582639, 4582624
37_1	2	1	1	8481643
42_1	2	1	1	5638107
45_1	2	1	1	5642079
51_1	2	1	1	4430955
63_1	2	1	1	4593176
74_1	2	1	1	4427529
78_1	3	2	2	4422184, 4422210
84_2	3	2	2	3694626, 3694730
89_1	2	1	1	6460987
205_1	2	1	1	4456340
206_1	2	1	1	6464695



(a)

(b)

Figure 4.5 Visual comparison of segment (2_1) in Navisworks (a) and its corresponding graph representation (b).

Figure 4.5 present a graphical analysis of clashes in the Piping–Ventilation. This figure focuses on segment (2_1), with the Navisworks view (right) showing a geometric clash between a pipe and mechanical equipment, and the Neo4j graph view (left) depicting the same clash as a two-node, one-edge subgraph. This comparison demonstrates how the graph-based workflow isolates individual clashes and identifies key components. Figure 4.6 represents cluster [Pipe, Mechanical Equipment] in Navisworks.

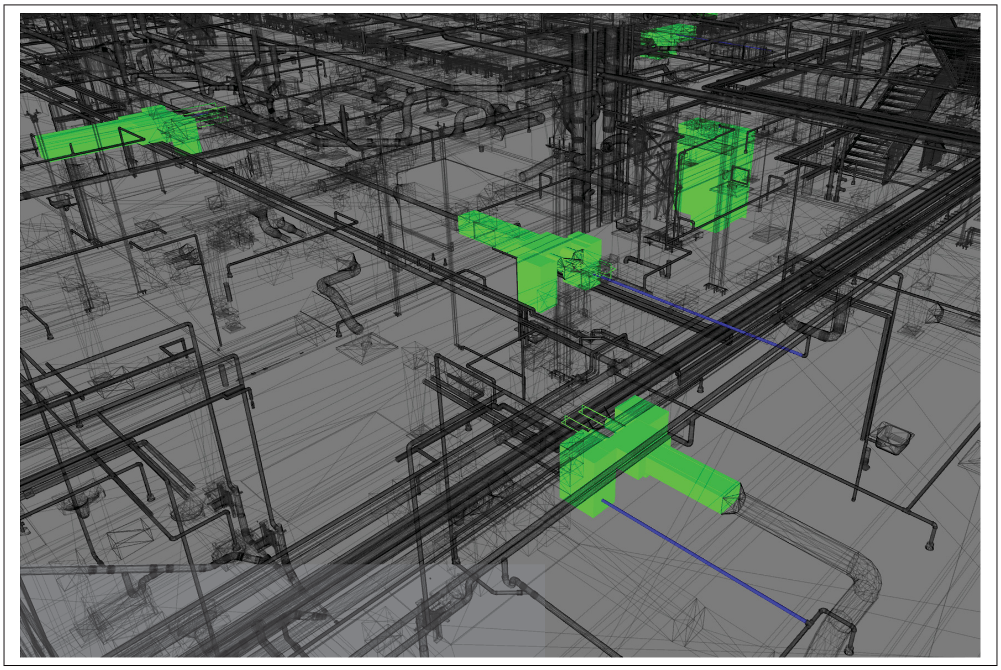


Figure 4.6 [Pipe, Mechanical Equipment] cluster in Navisworks.

4.2 Findings from case study 02: An Office

4.2.1 Project Presentation

This building, finalized in 2016, illustrates a contemporary and sustainable office development. The facility occupies an approximate area of 1,550 m² spread across two levels:

- Ground floor: 900 m²
- Second floor: 650 m².

4.2.2 Implementation of the Proposed Methodology (Structural vs Ventilation)

4.2.2.1 Initial Step

For this case study, we also used the default priority list from initial setup and made no changes to the assigned priority rankings (Table 3.1).

4.2.2.2 Clash detection

Clash detection performed on the federated model between the structural and MEP disciplines (piping and ventilation) identified a total of 173 clashes.

4.2.2.3 Graph creation

After importing the clash data into Neo4j, a graph-based data model was created in which each node represents a building component and each edge represents a clash relationship between two components. Attributes such as element type, discipline, and priority were assigned to each node, while edge properties included the clash identification number and the corresponding clash name.

The graph construction process resulted in a total of 185 nodes and 173 edges, representing all components and their respective clash relationships identified in the case study, as illustrated in Figure 4.7.

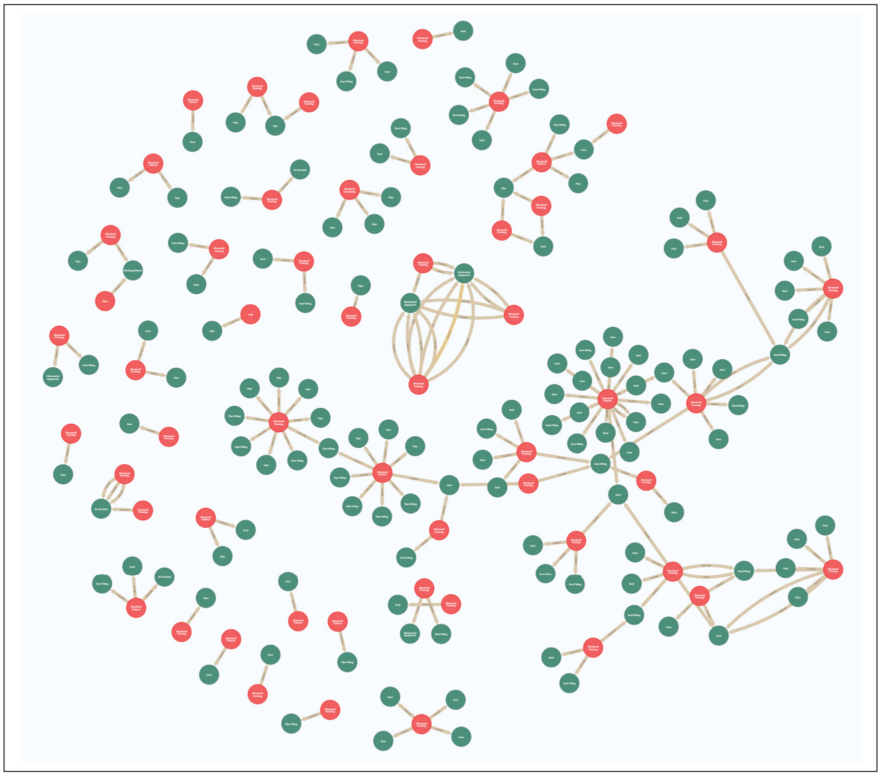


Figure 4.7 Graph representation of the office case (Structural vs MEP).

After building the graph in Neo4j, the WCC algorithm identified 33 isolated subgraphs, each consisting of elements connected by clashes and separate from other groups. The Louvain community detection algorithm was then applied, resulting in 14 distinct communities, each representing a localized area with multiple interconnected clashes.

A similarity analysis was conducted to group segments with similar element types, forming clusters with common clash characteristics. Table 4.6 presents the clustering results for Structural and MEP systems. Each row shows a pair of element types, the group IDs of similar subgraphs, and the total number of edges representing clashes between those components. For example, the first row shows several subgraphs with clashes between Structural Framing and Duct elements, totaling 12 edges. This indicates frequent conflicts between these components. The table highlights the number of clashes for each system pair and the segment IDs where they occur. To prioritize resolution, start with the cluster that has the highest number of edges, as it represents the most significant concentration of clashes.

Table 4.6 Similar segments and total edges between structural and MEP

Cluster	**Types**	**Similar Segments**	**Similar Segments Number**	**Total Edges**
1	[Structural Framing, Duct]	[6_1, 1_18, 15_1, 11_4, 36_1, 44_1, 45_1, 142_1, 148_1]	9	15
2	[Structural Framing, Mechanical Equipment]	[10_3, 10_4]	2	8
3	[Structural Framing, Pipe]	[23_2, 23_3, 28_1, 37_3]	4	4
4	[Structural Framing, Air Terminal]	[46_1]	1	4
5	[Structural Column, Duct]	[2_1, 29_1, 141_1]	3	3

4.2.2.4 Calculating MWVC

Table 4.7 summarizes the MWVC results for first cluster with the maximum number of edges between Structural Framing and Duct components. The MWVC algorithm identifies the minimal set of components whose modification would resolve all clashes within the local subgraph. A larger cover size indicates that clashes are more distributed across multiple components, whereas a cover size of one suggests that a single component is responsible for all conflicts in that group.

Segments 6_1 and 1_18 show relatively high cover sizes (4 and 3 respectively). However, after visual inspection in Navisworks, both segments were identified as false clashes. These conflicts are the result of incorrect bounding boxes rather than actual spatial interference; therefore, they do not require any design modification. Because these two segments do not represent real clashes, their MWVC results are not meaningful for decision-making.

Table 4.7 MWVC results for Structural Framing–Duct Similar Groups

Segment ID	**Nodes**	**Edges**	**Cover**	**Element ID**
6_1	5	4	4	988611, 1210718, 1211417, 988582
1_18	4	3	3	1132122, 946604, 946291
15_1	2	1	1	1222558
11_4	2	1	1	1044831
36_1	3	2	2	1294447, 1218222
44_1	2	1	1	1256377
45_1	2	1	1	1178305
142_1	2	1	1	1212527
148_1	2	1	1	1178518

For the remaining segments, the MWVC results correctly indicate the minimal set of components that should be adjusted.

4.3 Conclusion

This chapter presented the results of applying the proposed graph-based workflow to two real-world case studies with different scales. The hospital project demonstrated the ability of the approach to handle large datasets with a high number of clashes, revealing clear clusters of recurring similar clashes and identifying key components responsible for multiple conflicts. The comparison between manual grouping in Navisworks and proposed process flow in Neo4j showed alignment in terms of coordination logic, while the semi-automated process provided more consistent, and reproducible results. The MWVC analysis further supported decision-making by identifying minimal sets of components for modification within each clash segments.

The office building case study confirmed the applicability of the process flow to smaller projects. Across both case studies, the results highlight the capacity of graph-based modeling to capture component relationships and to support an organized approach to clash grouping. While visual inspection remains necessary to filter false positives, the findings demonstrate that the proposed workflow can support BIM coordinators by structuring complex coordination data and assisting in the identification of high-impact resolution actions. These results provide a basis for the discussion of limitations and future research directions presented in the following chapter.

CHAPTER 5

DISCUSSION

This chapter analyzes the findings from real-world case studies and places them within clash management in BIM-based multidisciplinary coordination. It first addresses key findings in relation to the research questions and hypotheses. Next, it compares these findings with existing BIM- and graph-based approaches in the literature. Finally, the chapter discusses the implications of the proposed artifact and outlines research limitations.

5.1 Discussion on the research questions, hypothesis and findings

This research aims to improve clash management efficiency in BIM-based multidisciplinary coordination by addressing two key challenges: the lack of systematic clash grouping and the absence of structured decision support for clash resolution. The study examines how graph-based methods can semi-automate the grouping of similar clashes and support resolution decisions to enhance coordination efficiency and reduce rework.

The first research question explored how considering dependencies among building components enables automatic grouping of related clashes. Case study results show that representing clashes as graphs, with components as nodes and clashes as edges, effectively captures these dependencies. Applying Weakly Connected Components and Louvain community detection algorithms allowed for automatic identification of localized clash groups that align with manual groupings by experienced BIM coordinators, while potentially improving efficiency and consistency. This supports hypothesis 1, which proposed that semi-automated grouping based on element types and dependencies improves clarity and efficiency over manual methods. The hospital case study demonstrated that proposed flow is able to divided clash dataset into meaningful clusters that reveal recurring interaction patterns between specific systems.

The second research question focused on strategies for prioritizing building components. Results show that combining component priority rankings with the MWVC algorithm helps identify lower-priority components in each clash segments. In many segments, the MWVC algorithm identified a single component whose modification can resolve multiple connected clashes, supporting hypothesis 2. This aligns with previous findings that clashes are interdependent and that modifying one element can resolve several conflicts at once Tabesh & Staub-French (2005).

The third research question examined how to implement the proposed strategy within a BIM-based coordination workflow. Case studies confirm that the artifact integrates into existing workflows by using standard clash detection outputs from Navisworks and converting them into graph representations with Neo4j. This workflow supports practical application without disrupting established coordination processes and meets the DSR requirement to evaluate artifacts under real conditions with actual project data.

5.2 Research limitations

Although the proposed graph-based methodology shows strong potential for improving clash grouping and resolution, several limitations must be noted:

- The methodology depends on the quality of the BIM models. Missing metadata, misclassified elements, or modeling inconsistencies directly influence the structure of the generated graph and affect clustering and MWVC results.
- The methodology addresses only hard clashes. It does not cover soft clashes, clearance issues, accessibility constraints, or rule-based and semantic conflicts.
- The full coordination cycle was not completed, as the suggested modifications were not implemented and re-evaluated through a new clash detection and workflow iteration.
- The proposed method has not been formally validated by professional BIM coordinators.
- The methodology assumes static input data and does not account for dynamic updates during the design process. In practice, BIM models evolve continuously, which may require repeated graph reconstruction and recalculation.

Despite these limitations, the study provides a foundation for future developments in automated clash management.

CONCLUSION AND FUTURE WORK

This thesis concludes by synthesizing the main findings from the literature review, artifact development, and case study evaluations. This chapter interprets the results in relation to the initial research questions and situates the contributions within the context of BIM-based multidisciplinary coordination. It summarizes key findings, connects them to the research objectives, discusses their impact on clash management practices, outlines study limitations, and suggests directions for future research.

This research followed the DSR methodology as formalized by Peffers *et al.* (2007), progressing through its key phases: problem identification and motivation, definition of objectives, design and development of the artifact, demonstration, evaluation, and communication. The proposed artifact was developed and demonstrated through application to two real-world case studies. However, the application of the DSR process in this thesis has limitation. The evaluation was conducted on only two case studies, which restricts the generalizability of the findings.

The literature review showed that studies has focused on detecting clashes rather than organizing them in a structured way for coordinated resolution. Existing approaches focus on detection or isolated resolution technique. This study responds to that need by proposing a approach to analyze and manage interconnected issues.

To address these limitations, this thesis enhances clash management through semi-automated clash grouping and decision support for resolution. The primary contribution is the use of graph-based modeling to explicitly represent building components and their clash relationships. By modeling components as nodes and clashes as edges, this approach captures inter-component relationships absent from traditional BIM workflows. This enables the identification of clash clusters and repeated interaction patterns, providing an alternative to manual grouping.

We verified the proposed process flow in real construction projects. The results show that the method follows the logic used by experienced BIM coordinators while reducing processing time and manual work. Case studies showed that the approach can handle clash datasets by dividing them into smaller group of clashes (segments) and cluster repetitive clashes.

In addition to clash grouping, this research proposes a process to identify a list of components that should be modified to resolve clashes using the MWVC algorithm. By integrating component priorities into the graph, the MWVC analysis identified minimal sets of components whose modification would resolve multiple connected clashes at once. Overall, the findings verified that integrating semi-automated clash grouping with proposing resolution strategies improves BIM coordination.

Although the proposed graph-based methodology shows strong potential for improving automated clash grouping and resolution, several limitations must be acknowledged. For example, the proposed methodology focuses exclusively on hard clashes involving geometric intersections and does not address soft clashes, clearance requirements, accessibility constraints, or semantic and rule-based conflicts, which remain important aspects of comprehensive BIM coordination.

Future research can expand on this work in several ways. One direction is to extend the graph-based approach to include soft clashes and clearance requirements. Integrating adaptive or learning-based priority schemes could allow the system to evolve based on project context or historical data. Incorporating artificial intelligence techniques to predict high-impact clash clusters early in the design process could enable more proactive coordination. Finally, validating the workflow across a wider range of project types, delivery methods, and organizational contexts would strengthen its generalizability and practical relevance management in BIM-based multidisciplinary coordination. By explicitly representing relationships among building components and combining semi-automated grouping with providing assistance for resolving similar clashes, the proposed approach enhances coordination efficiency. These contributions offer a meaningful step toward more intelligent clash management practices in the construction industry. Additionally, while the greedy approximation approach was selected for the MWVC calculation based on its computational efficiency and suitability for iterative workflows, the selection of the WCC and Louvain algorithms for clash grouping was not supported by a formal comparative analysis. Future work should therefore include a comprehensive assessment and comparison of alternative graph partitioning and community detection methods to establish whether the chosen algorithms represent the most effective solution for clash grouping in BIM-based coordination workflows.

APPENDIX I

GLOSSARY OF TERMS

Table-A I-1 Terminology used in the proposed methodology

Term	**Definition**
Cluster	A set of similar segments or disconnected subgraphs that share the same element types. It groups repeated clash patterns.
Disconnected Subgraphs (Islands)	Sets of nodes and edges that are connected to each other but disconnected from the rest of the network, representing independent components identified using the Weakly Connected Components (WCC) algorithm. Separates independent clash zones.
Edge	Represents a clash between two building elements.
Edge Property	Attributes associated with an edge, including clash name, clash point, and distance.
Group of clashes	Represents a set of clashes that are linked to one another within the coordination environment.
MWVC (Minimum Weighted Vertex Cover)	An optimization approach used to identify a set of lower-priority element IDs whose modification resolves all clashes in a cluster.
Node	Represents a building element.
Node Property	Attributes associated with a node, including element type, element ID, element priority, and element path.
Priority	A numerical value indicating the ease of modifying an element; elements with lower priority values are modified first during clash resolution.
Segments	Subgraphs within each disconnected subgraph, identified using the Louvain community detection algorithm.

APPENDIX II

PRIORITY CLASSIFICATION OF BUILDING SYSTEMS

Table-A II-1 Relative priority rating for system and sub-system classification

Main system	Sub-system	Priority Value
Architecture	Walls	46
Architecture	Ceilings	45
Architecture	Floors	44
Architecture	Roofs	43
Architecture	Envelope	42
Architecture	Stairs & Ramps	41
Architecture	Interior fixtures	40
Structure	Columns	39
Structure	Beams	38
Structure	Slabs	37
Structure	Foundations & Footings	36
Structure	Bracing	35
Structure	Structural walls	34
Structure	Trusses & Connections	33
Civil	Water supply	32
Civil	Sanitary	31
Civil	Stormwater	30
Civil	Gas	29
Civil	Irrigation	28
Large Equipment	HVAC Equipment	27
Large Equipment	Elevators & Escalators	26

Table-A II-1 Relative priority rating for system and sub-system classification (continued)

Main system	**Sub-system**	**Priority Value**
Large Equipment	Generators Equipment	25
Large Equipment	Plumbing Equipment	24
Large Equipment	Fire Protection Equipment	23
Large Equipment	Electrical Equipment (panels, switchboards, transformers)	22
Large Equipment	Commercial Equipment	21
Plumbing	Sanitary drainage	20
Plumbing	Stormwater drainage	19
Plumbing	Domestic	18
Plumbing	Vent pipes	17
Plumbing	Gas pipes	16
Plumbing	Floor drains	15
Plumbing	Roof drains	14
Mechanical HVAC	Air Handling Units	13
Mechanical HVAC	Ductwork	12
Mechanical HVAC	Hydronics	11
Mechanical HVAC	Variable Air Volume (VAV) Boxes	10
Mechanical HVAC	Exhaust fans & Diffusers	9
Fire protection	Sprinkler main	8
Fire protection	Sprinkler heads	7
Fire protection	Sprinkler branches	6
Electrical	Cable Tray	5
Electrical	Conduits	4
Electrical	Lighting fixtures	3
IT	Telecom	2

Table-A II-1 Relative priority rating for system and sub-system classification (continued)

Main system	**Sub-system**	**Priority Value**
IT	Security systems	1

BIBLIOGRAPHY

Autodesk. [Whitepaper, Autodesk Interoperability Tools for Revit, accessed 27 February 2025]. (2017). Classification Systems and Their Use in Autodesk Revit: Managing the “I” in BIM.

Balaji, S., Swaminathan, V. & Kannan, K. (2010). An effective algorithm for minimum weighted vertex cover problem. *Int. J. Comput. Math. Sci*, 4(1), 34–38.

Bar-Yehuda, R. & Even, S. (1981). A linear-time approximation algorithm for the weighted vertex cover problem. *Journal of Algorithms*, 2(2), 198–203.

Beetz, J., van Berlo, L., de Laat, R. & van den Helm, P. (2010). BIMserver. org–An open source IFC model server. *Proceedings of the CIP W78 conference*, 8.

Blondel, V. D., Guillaume, J.-L., Lambiotte, R. & Lefebvre, E. (2008). Fast unfolding of communities in large networks. *Journal of statistical mechanics: theory and experiment*, 2008(10), P10008.

Bratanic, T. (2024). *Graph Algorithms for Data Science: With Examples in Neo4j*. Simon and Schuster.

Burnham, J. F. (2006). Scopus database: a review. *Biomedical digital libraries*, 3(1), 1.

Cao, D., Wang, G., Li, H., Skitmore, M., Huang, T. & Zhang, W. (2015). Practices and effectiveness of building information modelling in construction projects in China. *Automation in construction*, 49, 113–122.

Chandra, B. & Halldórsson, M. M. (2001). Greedy local improvement and weighted set packing approximation. *Journal of Algorithms*, 39(2), 223–240.

Chinowsky, P., Taylor, J. E. & Di Marco, M. (2011). Project network interdependency alignment: New approach to assessing project effectiveness. *Journal of Management in Engineering*, 27(3), 170–178.

Ciribini, A. L. C., Ventura, S. M. & Paneroni, M. (2016). Implementation of an interoperable process to optimise design and construction phases of a residential building: A BIM Pilot Project. *Automation in construction*, 71, 62–73.

Clarkson, K. L. (1983). A modification of the greedy algorithm for vertex cover. *Inf. Process. Lett.*

Das, R. & Soylu, M. (2023). A key review on graph data science: The power of graphs in scientific studies. *Chemometrics and Intelligent Laboratory Systems*, 240, 104896.

De Sordi, J. O. (2021). *Design science research methodology: theory development from artifacts*. Springer Nature.

Eastman, C. M. (2011). *BIM handbook: A guide to building information modeling for owners, managers, designers, engineers and contractors*. John Wiley & Sons.

Garey, M. R., Johnson, D. S. et al. (1990). A Guide to the Theory of NP-Completeness. *Computers and intractability*, 37–79.

Han, J., Lu, X.-Z. & Lin, J.-R. (2025). Unified Network-Based Representation of BIM Models for Embedding Semantic, Spatial, and Topological Data. *arXiv preprint arXiv:2505.22670*.

Han, S., Lee, S. & Peña-Mora, F. (2012). Identification and quantification of non-value-adding effort from errors and changes in design and construction projects. *Journal of Construction Engineering and Management*, 138(1), 98–109.

Harode, A. & Thabet, W. (2021). Investigation of machine learning for clash resolution automation. *EPiC Series in Built Environment*, 2, 228–236.

Harode, A., Thabet, W. & Gao, X. (2022). An integrated supervised reinforcement machine learning approach for automated clash resolution. *Construction Research Congress 2022*, pp. 679–688.

Hevner, A. R., March, S. T., Park, J. & Ram, S. (2004). Design science in information systems research. *MIS quarterly*, 75–105.

Hochba, D. S. (1997). Approximation algorithms for NP-hard problems. *ACM Sigact News*, 28(2), 40–52.

Hsu, H. & Wu, I. (2019). Employing simulated annealing algorithms to automatically resolve MEP clashes in building information modeling models. *ISARC. Proceedings of the International Symposium on Automation and Robotics in Construction*, 36, 788–795.

Hu, Y. & Castro-Lacouture, D. (2019). Clash relevance prediction based on machine learning. *Journal of computing in civil engineering*, 33(2), 04018060.

Hu, Y., Castro-Lacouture, D. & Eastman, C. M. (2019). Holistic clash detection improvement using a component dependent network in BIM projects. *Automation in Construction*, 105, 102832.

Hu, Y., Castro-Lacouture, D. & Eastman, C. M. (2020a). BIM-enabled clash resolution optimization based on clash dependent networks. *Construction Research Congress 2020*, pp. 668–677.

Hu, Y., Castro-Lacouture, D., Eastman, C. M. & Navathe, S. B. (2020b). Automatic clash correction sequence optimization using a clash dependency network. *Automation in Construction*, 115, 103205.

Hu, Y., Castro-Lacouture, D., Eastman, C. M. & Navathe, S. B. (2021). Component change list prediction for BIM-based clash resolution from a graph perspective. *Journal of Construction Engineering and Management*, 147(8), 04021085.

Hu, Y., Xia, C., Chen, J. & Gao, X. (2023). Clash context representation and change component prediction based on graph convolutional network in MEP disciplines. *Advanced Engineering Informatics*, 55, 101896.

Hu, Z. & Zhang, J. (2011). BIM-and 4D-based integrated solution of analysis and management for conflicts and structural safety problems during construction: 2. Development and site trials. *Automation in construction*, 20(2), 167–180.

Huang, Y. & Lin, W. (2019). Automatic Classification of Design Conflicts Using Rulebased Reasoning and Machine Learning-An Example of Structural Clashes Against the MEP Model. *ISARC. Proceedings of the International Symposium on Automation and Robotics in Construction*, 36, 324–331.

Ismail, A., Nahar, A. & Scherer, R. (2017). Application of graph databases and graph theory concepts for advanced analysing of BIM models based on IFC standard. *Proceedings of EGICE*, 161.

Jonassen, D. H. (1997). Instructional design models for well-structured and III-structured problem-solving learning outcomes. *Educational technology research and development*, 45(1), 65–94.

Karp, R. M. (2009). Reducibility among combinatorial problems. In *50 Years of Integer Programming 1958-2008: from the Early Years to the State-of-the-Art* (pp. 219–241). Springer.

Khanzode, A., Fischer, M. & Reed, D. (2008). Benefits and lessons learned of implementing building virtual design and construction (VDC) technologies for coordination of mechanical, electrical, and plumbing (MEP) systems on a large healthcare project. *Journal of Information Technology in Construction (ITcon)*, 13(22), 324–342.

Kitchenham, B. et al. (2004). Procedures for performing systematic reviews. *Keele, UK, Keele University*, 33(2004), 1–26.

Korman, T. M. & Tatum, C. (2001). Development of a Knowledge-Based System to Improve Mechanical.

Korman, T. M., Fischer, M. A. & Tatum, C. (2003). Knowledge and reasoning for MEP coordination. *Journal of Construction Engineering and Management*, 129(6), 627–634.

Korman, T. M., Simonian, L. & Speidel, E. (2008). Using building information modeling to improve the mechanical, electrical, and plumbing coordination process for buildings. In *AEI 2008: Building Integration Solutions* (pp. 1–10).

Lambiotte, R., Delvenne, J.-C. & Barahona, M. (2008). Laplacian dynamics and multiscale modular structure in networks. *arXiv preprint arXiv:0812.1770*.

Lee, G. et al. (2014). Parallel vs. sequential cascading MEP coordination strategies: A pharmaceutical building case study. *Automation in Construction*, 43, 170–179.

Leite, F. L. (2019). *BIM for design coordination: A virtual design and construction guide for designers, general contractors, and MEP subcontractors*. John Wiley & Sons.

Lin, W. Y. & Huang, Y.-H. (2019). Filtering of irrelevant clashes detected by BIM software using a hybrid method of rule-based reasoning and supervised machine learning. *Applied Sciences*, 9(24), 5324.

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), 8.

Lin, Y.-C., Lin, C.-P., Hu, H.-T. & Su, Y.-C. (2018). Developing final as-built BIM model management system for owners during project closeout: A case study. *Advanced Engineering Informatics*, 36, 178–193.

Love, P. E., Edwards, D. J., Smith, J. & Walker, D. H. (2009). Divergence or congruence? A path model of rework for building and civil engineering projects. *Journal of performance of constructed facilities*, 23(6), 480–488.

March, S. T. & Smith, G. F. (1995). Design and natural science research on information technology. *Decision support systems*, 15(4), 251–266.

Meem, T. & Iordanova, I. (2022a). Clash avoidance in BIM-based multidisciplinary coordination: a literature overview. *Creative construction conference*.

Meem, T. M. & Iordanova, I. (2022b). Overcoming obstacles in BIM-based multidisciplinary coordination: a literature overview. *Canadian society of civil engineering annual conference*, pp. 637–654.

Mehrbod, S., Staub-French, S., Mahyar, N. & Tory, M. (2019a). Beyond the clash: investigating BIM-based building design coordination issue representation and resolution. *Journal of Information Technology in Construction*, 24.

Mehrbod, S., Staub-French, S., Mahyar, N. & Tory, M. (2019b). Characterizing interactions with BIM tools and artifacts in building design coordination meetings. *Automation in Construction*, 98, 195–213.

Mokhtar, A., Bédard, C. & Fazio, P. (2000). Collaborative planning and scheduling of interrelated design changes. *Journal of Architectural Engineering*, 6(2), 66–75.

Newman, M. E. (2010). Networks: an introduction. Oxford university press.

Newman, M. E. & Girvan, M. (2004). Finding and evaluating community structure in networks. *Physical review E*, 69(2), 026113.

Pärn, E., Edwards, D. & Sing, M. C. (2018). Origins and probabilities of MEP and structural design clashes within a federated BIM model. *Automation in Construction*, 85, 209–219.

Peffers, K., Tuunanen, T., Rothenberger, M. A. & Chatterjee, S. (2007). A design science research methodology for information systems research. *Journal of management information systems*, 24(3), 45–77.

Radke, A., Wallmark, T. & Tseng, M. (2009). An automated approach for identification and resolution of spatial clashes in building design. *2009 IEEE International Conference on Industrial Engineering and Engineering Management*, pp. 2084–2088.

Riley, D. R., Varadan, P., James, J. S. & Thomas, H. R. (2005). Benefit-cost metrics for design coordination of mechanical, electrical, and plumbing systems in multistory buildings. *Journal of Construction Engineering and Management*, 131(8), 877–889.

Tabesh, A. R. & Staub-French, S. (2005). Case study of constructability reasoning in MEP coordination. *Construction Research Congress 2005: Broadening Perspectives*, pp. 1–10.

Tranfield, D., Denyer, D. & Smart, P. (2003). Towards a methodology for developing evidence-informed management knowledge by means of systematic review. *British journal of management*, 14(3), 207–222.

Vom Brocke, J., Hevner, A. & Maedche, A. (2020). Introduction to design science research. In *Design science research. Cases* (pp. 1–13). Springer.

Wang, J., Wang, X., Shou, W., Chong, H.-Y. & Guo, J. (2016). Building information modeling-based integration of MEP layout designs and constructability. *Automation in construction*, 61, 134–146.

Wang, L. & Leite, F. (2012). Toward process-aware building information modeling for dynamic design and management of construction processes. *Proc., 19th Annual Workshop of the European Group for Intelligent Computing in Engineering, EG-ICE, Munich, Germany*.

Wang, L. & Leite, F. (2013). Knowledge discovery of spatial conflict resolution philosophies in BIM-enabled MEP design coordination using data mining techniques: a proof-of-concept. In *Computing in Civil Engineering (2013)* (pp. 419–426).

Wang, L. & Leite, F. (2016a). Formalized knowledge representation for spatial conflict coordination of mechanical, electrical and plumbing (MEP) systems in new building projects. *Automation in construction*, 64, 20–26.

Wang, L. & Leite, F. (2016b). Process knowledge capture in BIM-based mechanical, electrical, and plumbing design coordination meetings. *Journal of Computing in Civil Engineering*, 30(2), 04015017.