

*Article*

## Developing an Automated System for Checking the Strut Arrangement in Deep Excavation

Pisal Nov<sup>a</sup>, Vachara Peansupap<sup>b,\*</sup>, and Tanit Tongthong<sup>c</sup>

Department of Civil Engineering, Faculty of Engineering, Chulalongkorn University, Phayathai Rd., Wangmai, Pathumwan, Bangkok 10330, Thailand

E-mail: <sup>a</sup>Pisal.No@student.chula.ac.th, <sup>b</sup>pvachara@chula.ac.th (Corresponding author), <sup>c</sup>tanit.t@chula.ac.th

**Abstract.** At the early stage of the project, the clash problem in the strut arrangement was reviewed by using two-dimensional construction drawings. This practice was time-consuming and error-prone. Moreover, some 3D construction software could detect clashes between the structural strut and permanent building structures. But they were still limited in this arrangement including (1) decision of strut arrangement relying on human judgment; (2) unable to distinguish between unreal clash and real clash in the strut arrangement based on work practice, and (3) unable to present overview of all clash detections from each strut at one time. To solve these problems, this paper integrated rule-based approach with Building Information Modelling (BIM) to develop an automated system for detecting and solving clashes of the structural strut. The methodology primarily developed three main modules of an automated system by using rule-based approach. Then, these modules applied with BIM software, visual programming and spreadsheet for generating, checking and solving the clashes. To proof the efficiency, a case study of office basement construction was applied to test the system. As a result, this system could improve the strut arrangement work by reducing the human judgement. Moreover, the system has shortened the time of the strut arrangement by only specifying on the real clash problems. Last, this system also could provide better visualization of clashes.

**Keywords:** Strut arrangement, clash detection, building information modelling, deep excavation.

ENGINEERING JOURNAL Volume 25 Issue 1

Received 27 January 2020

Accepted 1 December 2020

Published 31 January 2021

Online at <https://engj.org/>

DOI:10.4186/ej.2021.25.1.99

## 1. Introduction

During the deep excavation work, the strutting system is a key supporting system that could prevent the failures of retaining walls and the accidents at the construction site [1]. In this strutting system, it is a structure that consists of strut, waler, knee, and kingpost. Figure 1 shows the elements of the strutting system. This structure is usually developed by concerning three main issues. First, the stability of the structure mainly focuses on the element size and spacing number between elements and elements of the strutting system. Moreover, it requires to monitor between the measured and designed axial forces at the construction site which could prevent the failure of the structure [2]. Next, the safety issue of the structure should be carefully designed to prevent the accident rates at the construction site. Since strut members used for workers' access and egress, the accidents due to falls from height and being trapped or struck by moving objects have become the most common causes of fatalities in Hong Kong [3]. Third, many constructible problems between structural strut and permanent building structures should not overlook in the strut arrangement. These problems still occur at the construction site and require having a review at the initial stage of the project. In short, although the strutting system is carefully designed and developed for protecting the adjacent buildings and providing safety for the worker, this system especially the structural strut should be properly checked in the arrangement.

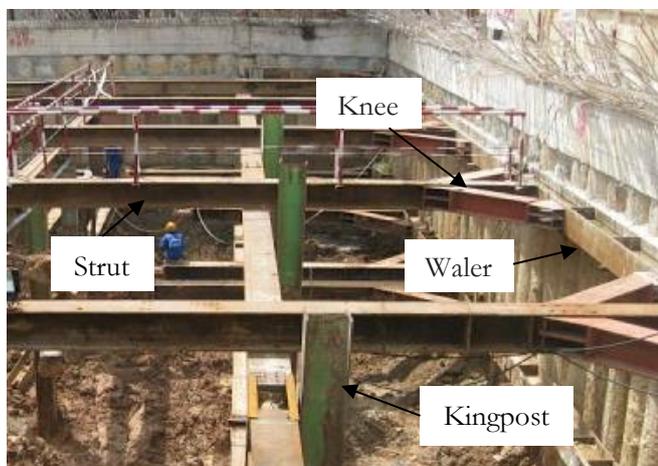


Fig. 1. Elements of strutting system.

Previous studies encountered some constructible problems of the structural strut at the construction site. First, inadequate space of strut was one of four main problems that contributed to the failure of the lateral bracing issue [4]. Moreover, this problem occurred in the deep excavation of Taiwan projects. Second, the strut levels were too close to slab concrete of the tunnel and there was very little clearance to work [5]. Moreover, they also found potential clashes between the strut and permanent building structures like columns or walls. In another constructible problem, insufficient horizontal space of struts could influence the access of workers,

supplies, and equipment [6]. In short, some constructible problems of structural strut were identified by past studies such as (1) Inadequate space among struts, (2) Insufficient space of horizontal and transverse struts, and (3) Clashes between the strut and permanent building structures. Thus, there were many types of constructible problems within the structural strut and could influence the time and cost of the excavation work.

Amount the above constructible problems, clash is an important problem that often occurs in the strut arrangement. The excavation work of the underground Central Artery Tunnel in downtown Boston found that the column was conflicted with the structural strut during the construction stage [5]. Moreover, there were some other potential clashes between the structural strut and permanent building structures like columns, walls, slabs, footings, and beams. During pre-construction phase, the architectural work sometimes changes in the layouts of underground structures due to new requirement of the owner. Then, this change could influence on the structural strut by modifying the strut location. Since the modification work conducted during the pre-construction of the structural strut, it has delayed the time of subsequent work operation and impacted on the fixing cost with the owner [3]. Therefore, the structural strut should properly arrange by identifying and solving any clash problems at the initial stage of the project.

## 2. Problem Statement

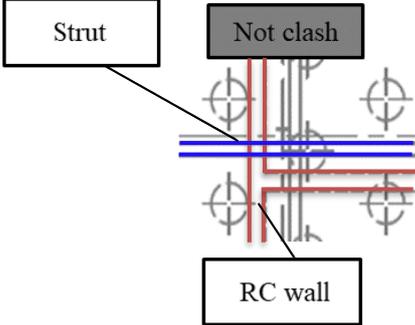
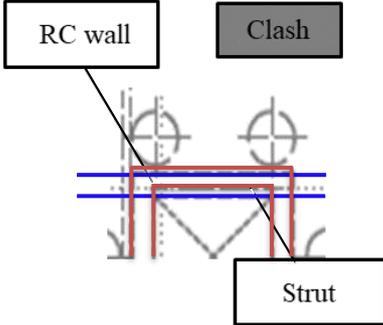
There are three main reasons that are still limit in traditional strut arrangement. First, the arrangement of strut is based on human judgement of clash problems between the structural strut and permanent building structures. This judgement uses the knowledge of strut arrangement which is obtained by many years' experience. Different engineers usually have different levels of knowledge for detecting and solving the clash problems of structural strut [6]. Some of them are still lack of experience to identify and solve the possible clash problems in the strut arrangement whereas the other engineers have much working experience and could give a reasonable answer to the problem without further analysis. To raise the emphasis on education and training of the strut arrangement, it would be another positive method to increase the skill, but it would take more time and cost in this investment. Thus, to avoid depending on the individual experience of engineer, a knowledge for identifying and solving the clash problems in the strut arrangement should be collected and developed into a system. Moreover, this system also could overcome a common method that still uses two-dimensional construction drawings to identify the clash problems of the structural strut [7]. This arrangement method of strut is time-consuming and error-prone.

Second, another problem has occurred with the distinction between unreal clash and real clash in the strut arrangement based on real practice. Although the available 3D construction software (ie. Autodesk Revit and

Navisworks Software) could simulate the construction process, check the clash and tolerance problems between strut and permanent building structures. However, these software applications still can't distinguish between unreal clash and real clash. For example, the conflict between strut and RC wall was considered as a real clash when it was found in the same direction (parallel direction). Moreover, the conflict between strut and wall was

considered as an unreal clash when it was found in a different direction. Table 1 shows unreal clash and real clash between strut and RC wall. Thus, when the proposed system is able to identify between unreal clash and real clash, the strut modification work could spend less time which only requires to focus on the real clash.

Table 1. An example of the unreal clash and real clash between strut and wall based on real practice.

Unreal clash	Real clash
Strut and wall are in the different direction	Strut and wall are in the same direction
<p><b>Case 1:</b></p> 	<p><b>Case 2:</b></p> 

Third, some available 3D construction software could provide better visualization of clash problems in a 3D model. However, these software applications still have a limitation to support the strut arrangement. They could display the clash problem one by one and is not able to display all clash occurrences on each strut at the same time. Thus, the engineer can't know the suitable location for moving the strut in order to solve the clash problems because the movement of strut may affect on other strut elements in the arrangement. Table 2 shows the difference between the available construction software and the proposed idea in clash detection. In short, human judgement of clash, unreal clash and real clash distinction and clash visualize limitation are still inefficiencies for identifying and solving the clash problem in the strut arrangement. This work should be improved by advanced construction technology like Building Information Modelling (BIM).

Building Information Modelling (BIM) is a digital representation of the facility physical characteristics. The facility information could use as a knowledge for decision-making during its life cycle from inception onward [8]. With its powerful capability for visualization and interoperability, BIM allows for the development of intelligent virtual building prototypes and the exchange of information throughout the project lifecycle. Integrated with systematically organized data and information, a highly collaborative work process is enabled to facilitate the communication and decision-making among different project participants. Thus, an inefficiency process is expected to be minimized and a better value will be delivered by using BIM [9]. In short, Building Information

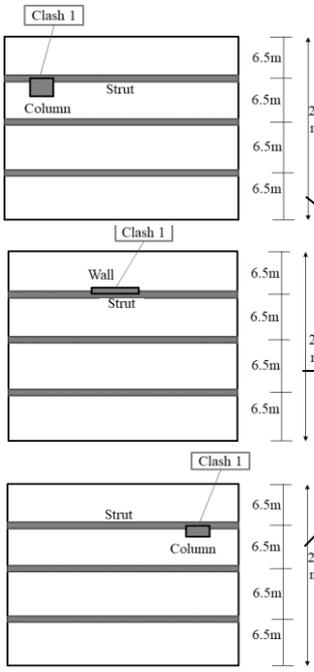
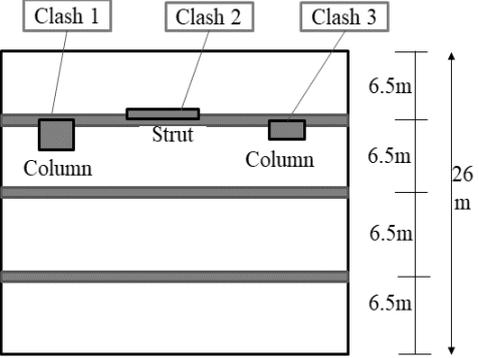
Modelling (BIM) can visualize and solve the clash problems between the structural strut and permanent building structures at the early stage of the project.

Building Information Modelling has been greatly known in the construction field for many years. This technology could integrate design and information for solving the inefficient process. This BIM technology can use to have early detection and allow the construction to proceed in a safe manner. Previous studies focused on the deep excavation issue with BIM technology. First, a framework with a web-based method was developed for analyzing the excavation alternatives in building construction projects [10]. This framework provided early cost estimates and schedules based on a probabilistic estimating and scheduling method and generated 4D simulations of the excavation method. Next, another research study developed an approach of BIM-based code compliance checking of deep foundation construction [11]. The control of code compliance at the design stage was an important method for ensuring safety in the deep foundation and reducing personal injury and property loss. Third, there was a study of the deep foundation pit project in Changsha Zhongqing Square [12]. This study used Revit software for establishing a 3D model of the excavation pit supporting system. The structural collision checking, simulation of the construction process and deepening design of nodes were carried out to guide to building construction. An automated safety risk recognition process was developed based on BIM which composed of three parts including (1) building risk database, (2) relation analysis between engineering information and safety risks, and (3) automated safety risk recognition mechanism in

the BIM platform [9]. Recently, the object t-level in design was developed by integrating rules of modular coordination with BIM [13]. The rules of modular coordination consisted of (1) reference system, (2) preferred sizes, (3) alignment system, (4) 5 mm rule/tolerance, (5) joint details. As a result, it could assist the modeller in placing and alignment of building elements within the modular reference frame, reference lines/planes. Last, some other studies related to BIM in the deep excavation are explained in Table 3. Therefore, previous BIM researches in deep excavation have been

proposed to solve two main aspects. First, the research studies focused on stability and safety during the deep excavation. Moreover, BIM was used as a model that integrated with geotechnical data for designing and monitoring the soil excavation. Second, the research on BIM related to the deep excavation focused on the visualization in the 3D model. In conclusion, there are many research studies between BIM technology and deep excavation. However, there is not many studies of BIM application on strut arrangement in excavation work that could identify and solve the clash problems.

Table 2. Difference between the available construction software and proposed idea in clash detection.

	<b>The clash detection by current available software</b>	<b>The proposed idea of clash detection in this study</b>
The display ability of the clash occurrence	Display the clash problem one by one	Display all clash occurrences on each strut at the same time for making decision on strut arrangement
Picture of clash occurrence		

The utilization of BIM with a strutting system has not much focused on previous studies. A case study of BIM can improve the design and construction productivity and aid geotechnical design in deep excavation projects in Jakarta, Indonesia [18]. Due to this study, BIM was primarily built by a Revit model to understand the complex geometry of the permanent building structures before designing the scheme of supporting system. Second, the strut locations were easily understood by the BIM model. Moreover, The BIM model not only allowed the engineers to see the strutting system in a 3D model but also allowed the engineers to have a macro view of the overall design, identify the critical areas and produce a more optimized design. Thus, the BIM model has been a critical technology to improve the overall design and construction productivity. Another study developed an ontology-based system for construction risk knowledge

management in the BIM environment [7]. In this system, some general risk rules had been predefined by the knowledge base. When a specific construction object in the BIM model was selected, the general risk query rules were customized with the relevant information. Thus, this system could help the user (1) identify the construction processes and the potential risks of monitoring objects as strut, (2) analyze the risk factors, and (3) take the risk precautions to prevent the occurrence of an accident according to an identified risk. Although previous studies focused on some areas of strutting system with BIM concept, the study of constructible problems like clashes still required to improve for supporting the review in the strut arrangement. Moreover, by using BIM, it will provide an overview design of structural strut in the 3D model environment, and early identification and solution of clash problems. In short, some 3D construction software could

detect clashes between the structural strut and permanent building structures. But they were still limited in this arrangement including (1) decision of strut arrangement relying on human judgment; (2) unable to distinguish between unreal clash and real clash in the strut arrangement based on work practice, and (3) unable to present overview of all clash detections from each strut at the one time. Therefore, the objective of this paper integrated rule-based approach with Building Information Modelling (BIM) to develop an automated system for detecting and solving clashes of the structural strut.

Table 3. Previous studies related to BIM in the deep excavation.

Author, Year	Research description
Lu, Wu and Hsiung, [14]	Applied a real-world case study to integrate the 3D building model, and excavation model. The results of ground surface settlement analysis, measurement, and monitoring data into one system could use to accurately execute environmental impact assessments.
Wu, Lu and Hsiung, [15]	Developed a Building Information Modelling (BIM)-based monitoring system to integrate and visualize the monitoring data for risk assessments during urban deep excavation projects. The system accurately visualized the location of monitoring instruments in the BIM system and provided the data requirement to assess the ground settlement by using numerical analysis.
Lin et al., [16]	Developed a BIM-based approach for settlement monitoring. This study could determine possible risks by comparing the monitoring data and warning levels. Moreover, different methods were proposed to control the impact which caused by ground settlement.
Szwarkowski and Pilecka, [17]	Studied the use of BIM in the modeling of high building “Mogilska Tower” in Cracow. The high building model was developed and applied with the data of geotechnical parameters for checking the settlement impact and evaluating the influence of neighbouring buildings.

### 3. Methodology

To achieve the purpose of this research study, the methodology was primarily conducted by collecting the information of the strut arrangement from a couple of experts' interviews. These experts were senior engineers of subcontractor companies in Thailand. Moreover, they had at least 15 years of experience in deep excavation work of building construction. The information result of the strut arrangement was presented in another paper elsewhere and this paper only focused on the system development. In this paper, it was begun by applying a rule-based approach for supporting of the system development. This approach is based on the IF-THEN statement that has input and output information. The approach developed into three main modules of a system such as strut generation, clash detection of strut, and automated strut modification. Then, each module was applied with BIM software (Autodesk Revit Software), visual programming (Dynamo Software) and spreadsheet (Microsoft Excel Software). Last, after developing the system, a case study of office basement construction was tested with the system development.

## 4. System Development

This section explains the process on developing the system. The system consists of rule-based generation, preliminary structural strut generation, modules for clash detection, and strut modification. The rule-based generation from experts was used to support the structural strut generation, modules for clash detection and strut modification. At first, the structural strut generation module aims to create the preliminary layout of structural strut elements both the transverse and horizontal direction. In addition, this module creates the layout structural strut elements in each level from information of 3D building information model. In this structural strut module, numbers of strut levels and strut quantities in horizontal and transverse directions were determined. Next, modules for clash detection were developed to detect the clashes between preliminary strut arrangement and 3D building structures. The result of clash detection will be used to next module. Last, the strut modification was designed to re-arrangement of the final structural strut. Thus, it could avoid the intersection between strut and building structures. The detail information of each modules in the system development was explained in the following section.

### 4.1. Rule-based Approach for Supporting System

Rule-based approach has been developed and implemented in the AEC industry for structural element design, building code checking, building performance checking, and safety planning. A rule is considered as a formal and popular way of “representing recommendations, directives, or strategies when the domain knowledge results from empirical associations developed through years of experience solving problems in an area” (Waterman, 1986). IF-THEN statements can

be used to express and organize the rule-sets with logic as human reasoning. IF links the condition for rule execution, including the facts and criteria; THEN is followed by the execution results. When the IF portion of a rule is satisfied by the facts, the action is specified by the THEN portion. In this study, the rule-based approach is applied to develop into three modules for generating, detecting, and solving the clash problems of strut arrangement.

The automated system basically attempted to solve the clash problems between structural strut and permanent building structures. This system was primarily developed by using the rule-based approach. The approach used the information which was obtained from the expert interview and developed into three main modules of the system. In the first module, the strut

generation determined the numbers of strut levels and quantities of horizontal and transverse struts. Moreover, this module could avoid the clash occurrence between (1) struts and footings, (2) struts and slabs. In the next module, after generating the initial structural strut by the first module, the detection of strut only checked the clash occurrence between the structural strut and permanent building structures like columns, beams and RC walls. In the last module, the automated strut modification relocated the strut to another place for solving the clash problems. Each module of the system was clearly explained in the following section. As a result, the final structural strut was obtained without the clash problems. Figure 2 shows a framework of an automated system development.

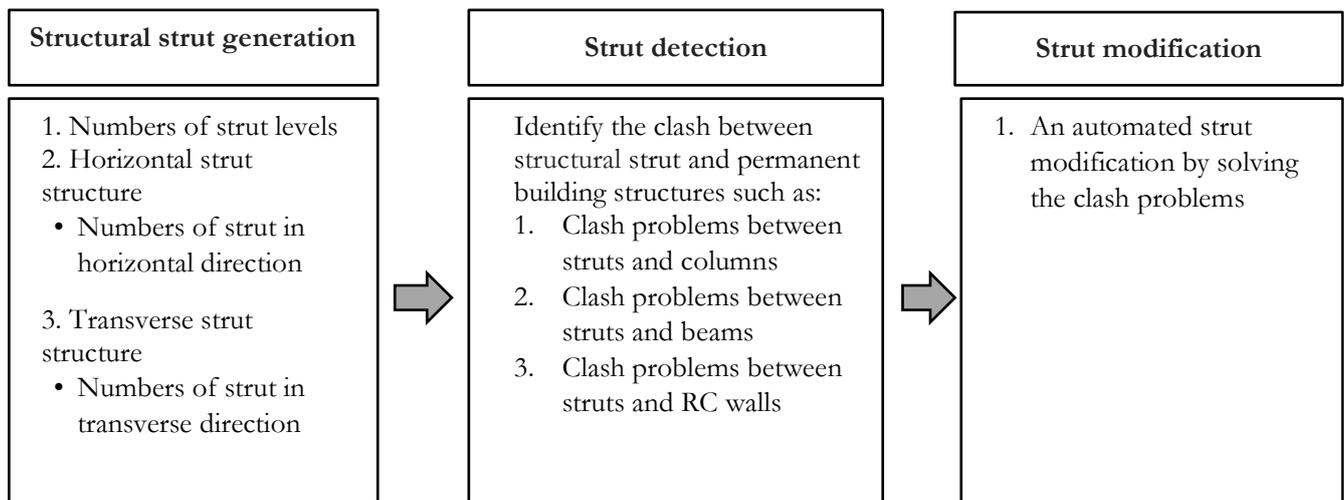


Fig. 2. A framework of an automated system development.

## 4.2. Modules of Structural Strut Generation

The first module was the generation of an initial structural strut. This module was divided into two main steps including (1) the determination of strut level numbers and (2) strut quantities in horizontal and transverse directions. Each step was developed by using rule-based approach that was based on the information of strut generation. Then, the system was developed by using Autodesk Revit and Dynamo Software.

### 4.2.1. Numbers of strut levels (vertical position)

In the first step, the numbers of strut levels were determined by using the input information from 3D model such as (1) depth number of excavation work and (2) value of all slab levels. Moreover, the input also needs to specify the strut size like 300mm, 350mm, 400mm. First, the depth number of excavation work was used to determine the material type of retaining wall whether concrete pile or sheet pile. If the concrete pile was selected as the material type of retaining wall, the first strut level could be equal or bigger than -2m. If the sheet pile was selected as the material type of retaining wall, the first strut

level could be equal or bigger than -1.5m. Next, the value of all slab levels was used to compare with the first strut level for checking whether they were different or same level. If the first strut level was not at the same level with the other slabs, it was selected as a trial strut level and continue to determine the first slab level. Last, the strut size allowed the user to select and determine the number of 1st and next strut level. In the process stage, the rule-based approach was applied to (1) determine the material type of retaining wall, (2) check the different between 1st and next strut level and all slab levels, (3) verify between the 1st and next strut level and all slab levels. When strut level exceeded the depth of excavation level, the determination process was stopped and the numbers of strut level were also determined as the result. Finally, the first step of module could specify the results of (1) the material type of retaining wall, (2) the number of 1st and next strut level, and (3) verification result of the 1st and next strut level with all slab levels. Figure 3 shows the flow chart information for determining the strut level numbers. After the numbers of strut levels are determined from the above process, this result is applied to determine the quantity of horizontal and transverse struts at each level.

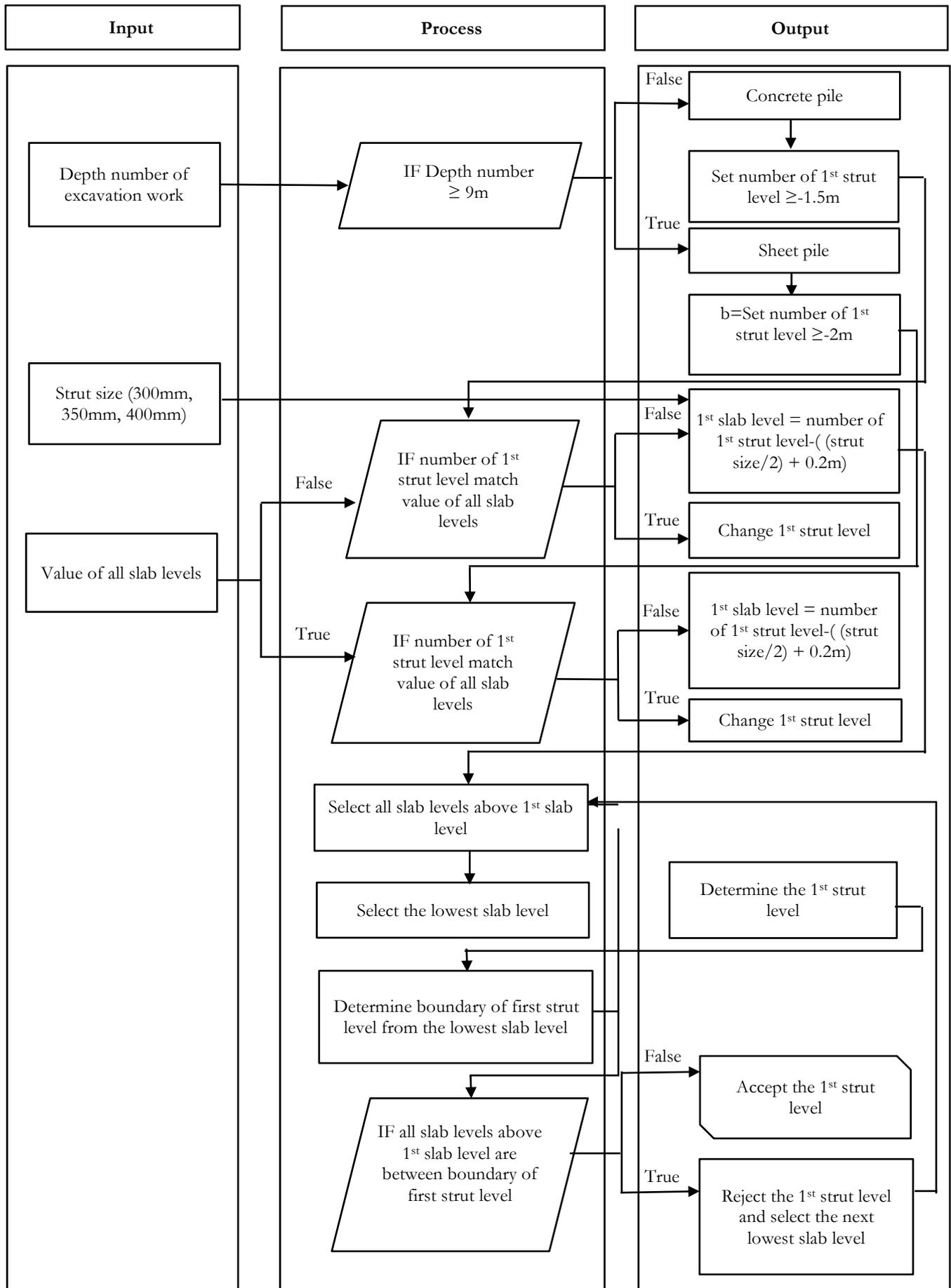


Fig. 3. Flow chart information for determining the strut level numbers.

4.2.2. Strut quantities in horizontal and transverse directions

In the second step, the strut quantities in horizontal and transverse directions were determined in each strut level that had found from the first step of this module. First, the input information was required to obtain from 3D model such as (1) retaining wall thickness, (2) the width and length of the retaining wall, and (2) specified spacing value of strut (i.e. 6m). First, transverse and horizontal lines of each level were determined by subtracting the width and length of retaining wall with the retaining wall thickness. Then the numbers of transverse and horizontal struts were determined by dividing the

transverse and horizontal lines with the specified spacing value of strut. The rule-based approach was applied to determine whether the length number in transverse direction is longer than the length number in horizontal direction or not. If the result was true, the vertical number of lines in transverse direction summed with the size number of struts. Last, the transverse and horizontal lines of each level combined with strut family type for generating the 3D models of transverse and horizontal struts in each level. Figure 4 shows the flow chart information for determining strut quantities in horizontal and transverse directions.

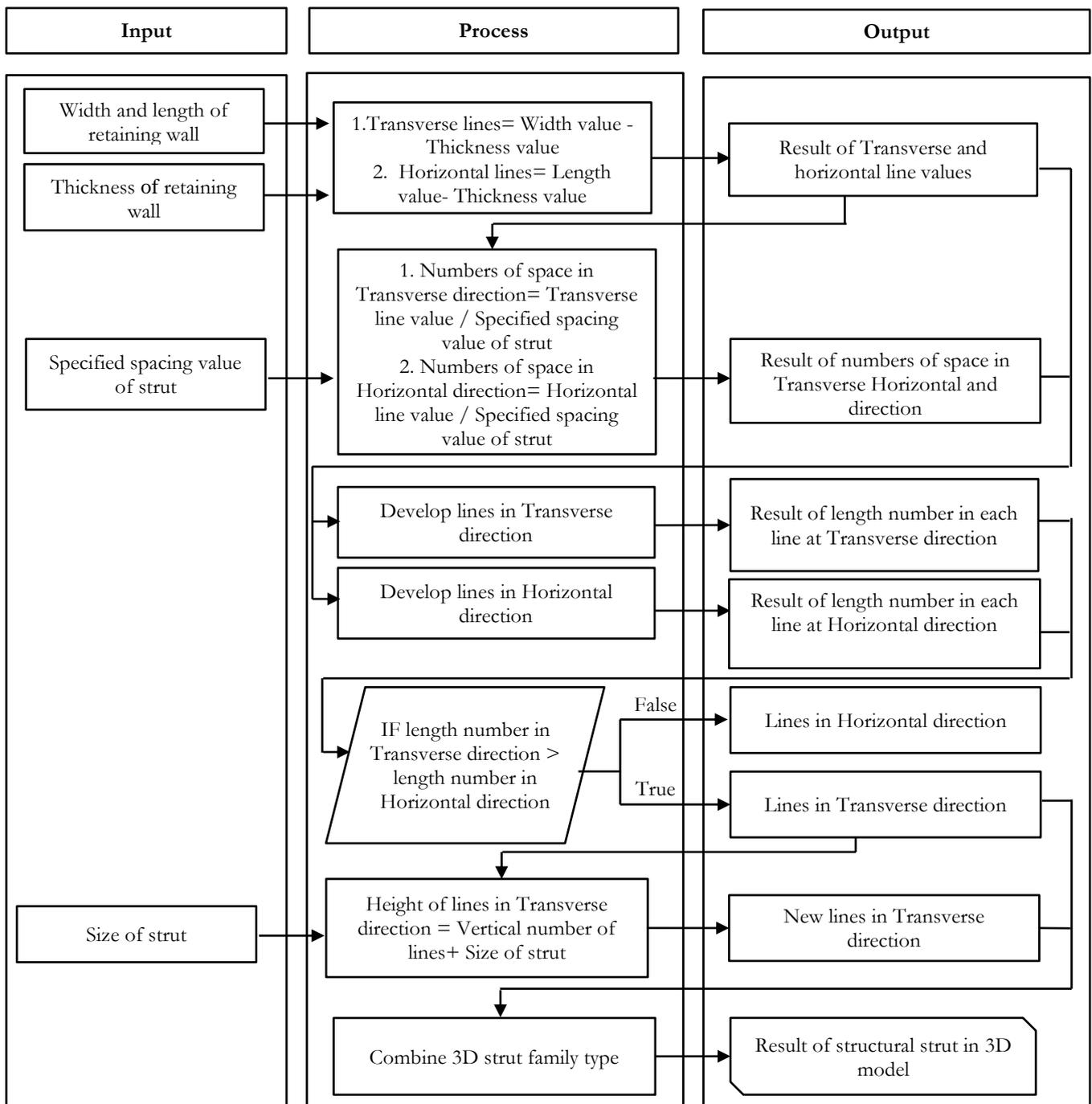


Fig. 4. Flow chart information for determining strut quantities in horizontal and transverse directions.

#### 4.2.3. Preliminary strut generation in 3D model

After the initial structural strut was determined and generated into a 3D model, this structural strut checked the clash between structural strut and permanent building structures. Due to the results of the strut generation at the first stage, this module could avoid to undertake the clash detection between (1) strut and slab, (2) strut and

foundation which has pile and footing. Table 4 shows the results of strut generation in 3D model between (1) strut and slab, (2) strut and foundation. In this study, the system development of clash detection was only responsible for checking between (1) strut and column, (2) strut and wall and (3) strut and beam. The detail information of each clash detection was explained in the following section.

Table 4. Strut generation between (1) strut and slab, (2) strut and foundation.

	Strut Plan View	Strut Section View
Strut and slab		
Strut and foundation		

#### 4.3. Modules for Checking Clash between Strut and Building Structures

This section explains the checking clash detection between strut and building structures. In depth excavation, the strut can be conflicted with several building structures such as column, wall and beam element. The concepts and algorithms were described below.

##### 4.3.1. Clash detection between strut and column structure

First, the clash detection between strut and column structure was developed by classifying into input, process,

and output. First, the input data were the strut elements and the column elements. Then, the detection process applied the rule-based approach to find the clash between both elements. As a result, the detected struts were determined and divided into transverse and horizontal directions. The detected struts and columns were also highlighted by adding color in the 3D model. Last, all the detected information saved in the spreadsheet including clash number, column type, column ID, strut type, strut ID, Point. X, Point. Y, Point. Z, embedded thickness value of clash, column width, column length. Figure 5 shows flow chart information of clash detection between strut and column structure.

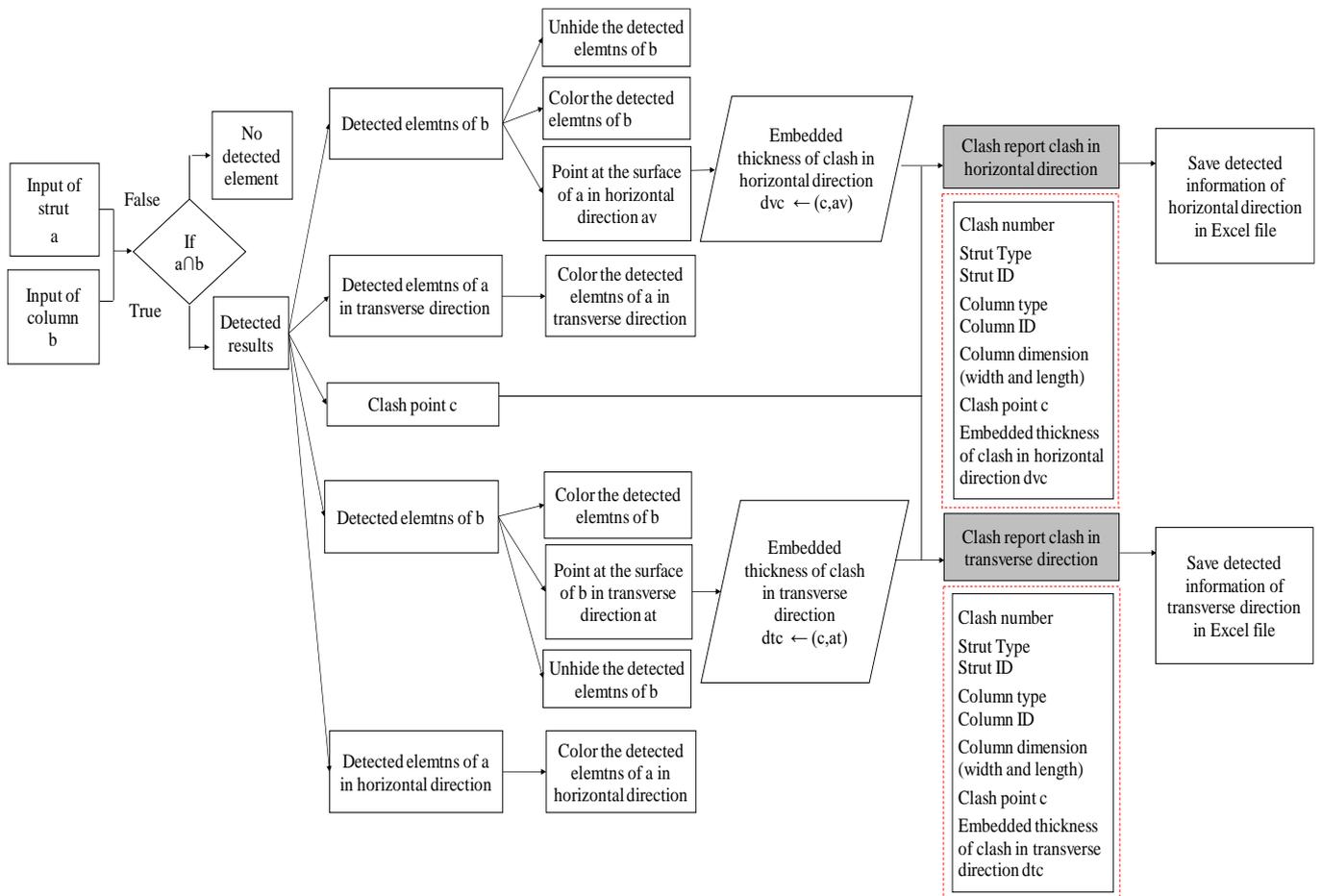


Fig. 5. Flow chart information of clash detection between strut and column structure.

4.3.2. Clash detection between strut and wall structure

Next, the clash detection between strut and wall structure was developed by classifying into input, process, and output. First, the input data were the strut elements and the wall elements. Moreover, the struts and walls were divided into transverse and horizontal directions. In the detection process, the rule-based approach was applied to find the clash between struts and walls in the parallel direction only (i.e. transverse strut Vs transverse wall or horizontal strut Vs horizontal wall). As a result, the

detected struts were determined and divided into transverse and horizontal directions. The detected struts and walls were also highlighted by adding color in the 3D model. Finally, all the detected information saved in the spreadsheet including clash number, wall type, wall ID, strut type, strut ID, Point. X, Point. Y, Point. Z, embedded thickness value of clash, wall width. Figures 6 and 7 show flow chart information of clash detection between struts and walls in transverse and horizontal directions.

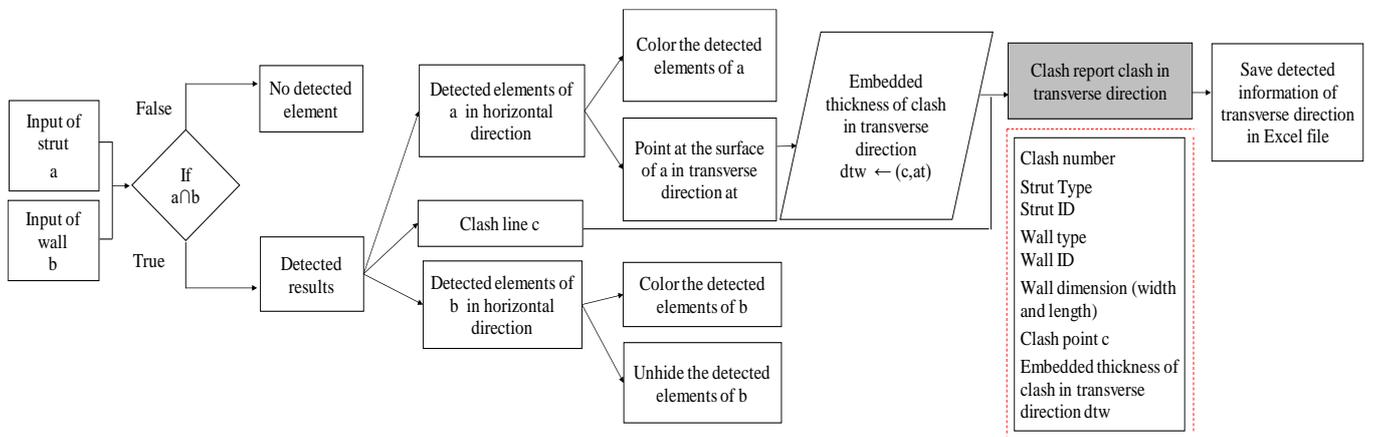


Fig. 6. Flow chart information of clash detection between struts and walls in the horizontal direction.

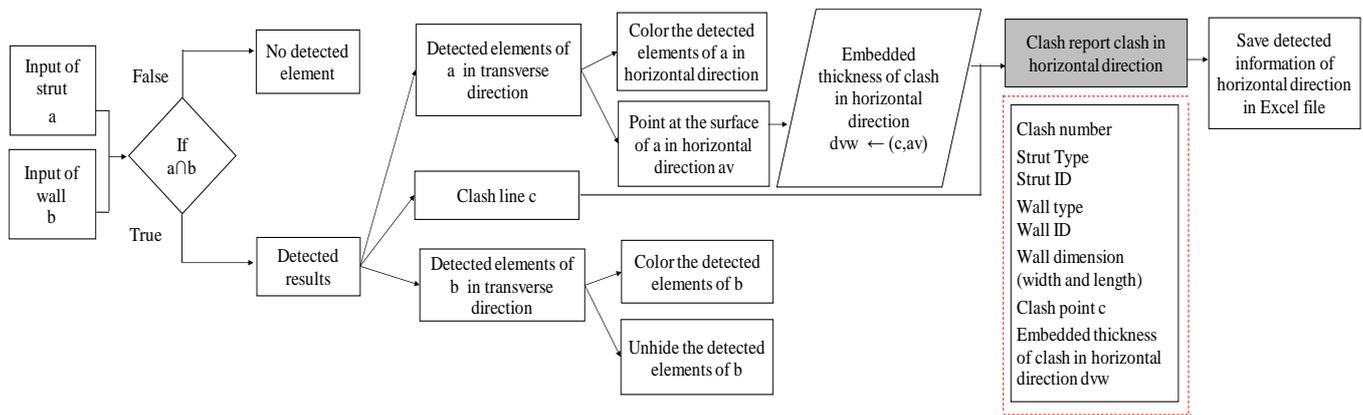


Fig. 7. Flow chart information of clash detection between struts and walls in the transverse direction.

#### 4.3.3. Clash detection between strut and beam structure

Last, the clash detection between strut and beam structure was developed by classifying into input, process, and output. First, the input data were the strut elements and the beam elements. Then, the detection process applied the rule-based approach to find the clash between both elements. As a result, the result of detected strut and beam elements, clash points, and clash views were

obtained from the transverse and horizontal directions. The detected struts and beams were also highlighted by adding color in the 3D model. Last, all the detected information saved in the spreadsheet including clash number, beam type, beam ID, strut type, strut ID, Point. X, Point. Y, Point. Z, width of beam. Figure 8 shows flow chart information of clash detection between strut and beam structure.

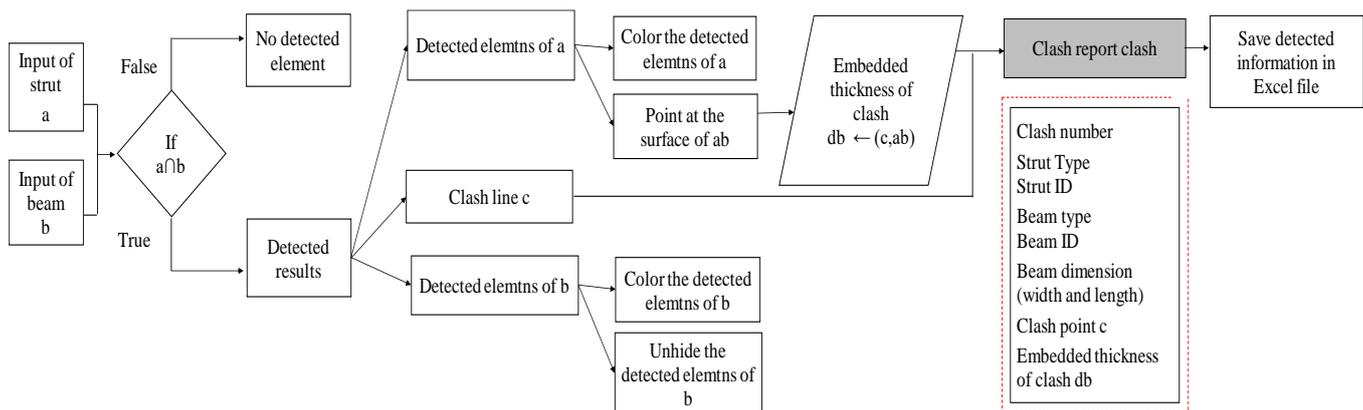


Fig. 8. Flow chart information of clash detection between strut and beam structure.

#### 4.4. Module of Strut Modification

After clash cases are detected by previous modules, the module of strut modification attempts to automatically solve the clash problems. The module of strut modification basically tried to solve the clash problem by moving the strut to another location. Due to the structural strut generation, it generated the strut above the floor, so the clash between strut and beam structure rarely occurred from the detection. Moreover, if the clash between strut and beam of ramp was occurred in the 3D model, this clash was not occurred in the real practice because the ramp beam is constructed after the removal of strut. Thus, in this study, the modification function only considered the clash between (1) strut and column, (2) strut and wall. Therefore, in this modification, the strut relocated either transverse or horizontal direction.

Initially, this modification primarily considered the characteristic of clashes between strut and permanent

building structures like beam, column, and wall structures. Then, it could have different algorithms for determining the embedded thickness value. The embedded thickness value is very important to move the strut to a new location for solving the clash problem. Moreover, a new modification value of strut also considers the working space for worker between strut and permanent building structures like wall and column by adding the embedded thickness value with allowance tolerance number around 300mm. Tables 5 and 6 show a characteristic of clashes between strut and column and algorithm for determining the new modification value of strut in horizontal and transverse directions. Table 7 shows a characteristic of clashes between strut and wall and algorithm for determining the new modification value of strut. Table 8 shows a characteristic of clashes between strut and beam and algorithm for determining the new modification value of strut.

Table 5. A characteristic of clashes between strut and column and algorithm for determining the new modification value of strut in the transverse direction.

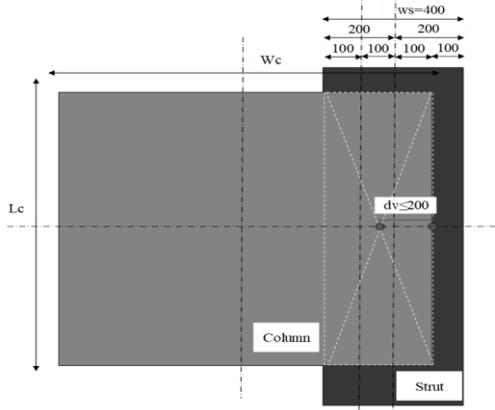
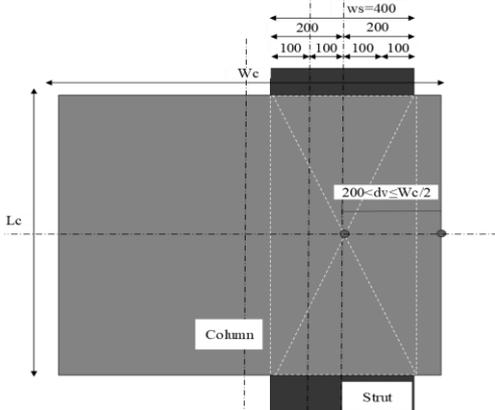
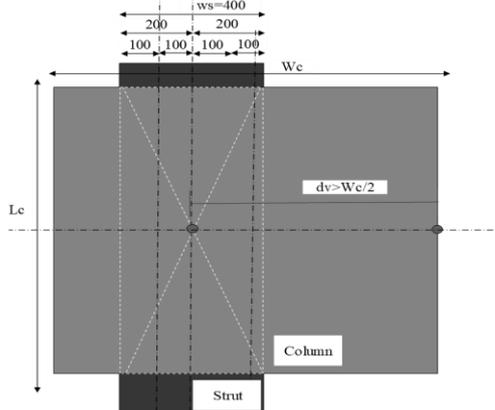
N	Number of characteristics	Algorithm of new modification value of strut	Characteristics of clashes between transverse strut and column in plan view
1	First characteristic of transverse strut	$M = (dv \times 2) + T$ <p>Where:</p> <ul style="list-style-type: none"> <li>• M is a modification value</li> <li>• dv is an embedded thickness value of clash in transverse direction</li> <li>• T is an allowance tolerance number</li> </ul>	
2	Second characteristic of transverse strut	$M = (dv - 200) + 400 + T$ <p>Where:</p> <ul style="list-style-type: none"> <li>• M is a modification value</li> <li>• dv is an embedded thickness value of clash in transverse direction</li> <li>• T is an allowance tolerance number</li> </ul>	
3	Third characteristic of transverse strut	$M = - ((Wc - dv) + 200 + T)$ <p>Where:</p> <ul style="list-style-type: none"> <li>• M is a modification value</li> <li>• Wc is a width value of column</li> <li>• dv is an embedded thickness value of clash in transverse direction</li> <li>• T is an allowance tolerance number</li> </ul>	

Table 6. A characteristic of clashes between strut and column and algorithm for determining the new modification value of strut in the horizontal direction (Continue).

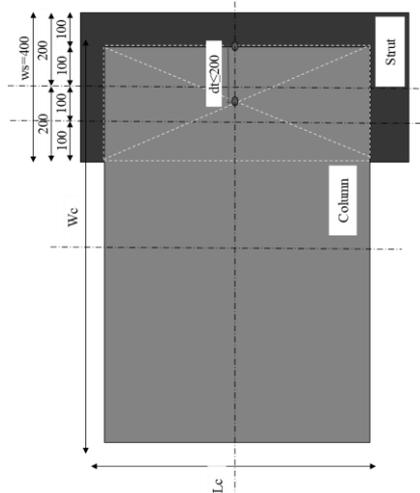
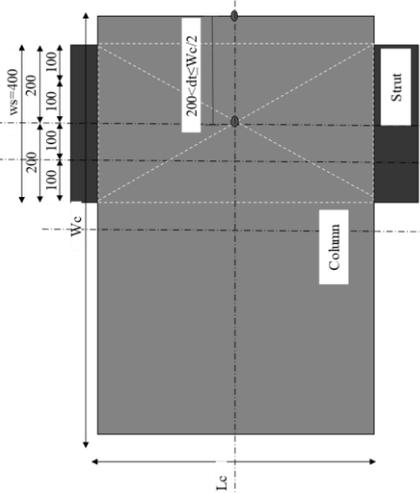
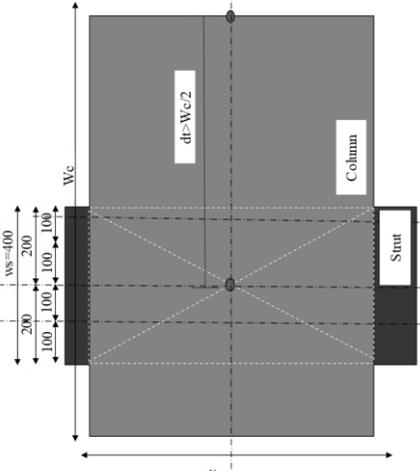
N	Number of characteristics	Algorithm of new modification value of strut	Characteristics of clashes between horizontal strut and column in plan view
4	First characteristic of horizontal strut	$M = (dt \times 2) + T$ <p>Where:</p> <ul style="list-style-type: none"> <li>• M is a modification value</li> <li>• dt is an embedded thickness value of clash in horizontal direction</li> <li>• T is an allowance tolerance number</li> </ul>	
5	Second characteristic of horizontal strut	$M = (dt - 200) + 400 + T$ <p>Where:</p> <ul style="list-style-type: none"> <li>• M is a modification value</li> <li>• dt is an embedded thickness value of clash in horizontal direction</li> <li>• T is an allowance tolerance number</li> </ul>	
6	Third characteristic of horizontal strut	$M = - ((Lc - dt) + 200 + T)$ <p>Where:</p> <ul style="list-style-type: none"> <li>• M is a modification value</li> <li>• Lc is a length value of column</li> <li>• dt is an embedded thickness value of clash in horizontal direction</li> <li>• T is an allowance tolerance number</li> </ul>	

Table 7. A characteristic of clashes between strut and wall and algorithm for determining the new modification value of strut in horizontal and transverse directions.

New modification value of strut in the transverse direction			
First characteristic	Second characteristic	Third characteristic	Fourth characteristic
$M = - ((400 - dv) \times 2) + T$ <p>Where:</p> <ul style="list-style-type: none"> <li>• M is a modification value</li> <li>• dv is an embedded thickness value of clash in transverse direction</li> <li>• T is an allowance tolerance number</li> </ul>	$M = - ((400 - dv) \times 2) + T$ <p>Where:</p> <ul style="list-style-type: none"> <li>• M is a modification value</li> <li>• dv is an embedded thickness value of clash in transverse direction</li> <li>• T is an allowance tolerance number</li> </ul>	$M = (dv \times 2) + T$ <p>Where:</p> <ul style="list-style-type: none"> <li>• M is a modification value</li> <li>• dv is an embedded thickness value of clash in transverse direction</li> <li>• T is an allowance tolerance number</li> </ul>	$M = (dv \times 2) + T$ <p>Where:</p> <ul style="list-style-type: none"> <li>• M is a modification value</li> <li>• dv is an embedded thickness value of clash in transverse direction</li> <li>• T is an allowance tolerance number</li> </ul>

Table 7. A characteristic of clashes between strut and wall and algorithm for determining the new modification value of strut in horizontal and transverse directions (Continue).

N	Number of characteristics	Algorithm of new modification value of strut	Characteristic picture of clashes between horizontal strut and wall
1	First characteristic	$M = - ((400 - dt) \times 2) + T$ <p>Where:</p> <ul style="list-style-type: none"> <li>• M is a modification value</li> <li>• dt is an embedded thickness value of clash in horizontal direction</li> <li>• T is an allowance tolerance number</li> </ul>	
2	Second characteristic	$M = - ((400 - dt) \times 2) + T$ <p>Where:</p> <ul style="list-style-type: none"> <li>• M is a modification value</li> <li>• dt is an embedded thickness value of clash in horizontal direction</li> <li>• T is an allowance tolerance number</li> </ul>	
3	Third characteristic	$M = (dt \times 2) + T$ <p>Where:</p> <ul style="list-style-type: none"> <li>• M is a modification value</li> <li>• dt is an embedded thickness value of clash in horizontal direction</li> <li>• T is an allowance tolerance number</li> </ul>	
4	Fourth characteristic	$M = (dt \times 2) + T$ <p>Where:</p> <ul style="list-style-type: none"> <li>• M is a modification value</li> <li>• dt is an embedded thickness value of clash in horizontal direction</li> <li>• T is an allowance tolerance number</li> </ul>	

Table 8. A characteristic of clashes between strut and beam and algorithm for determining the new modification value of strut.

N	Number of characteristics	Algorithm of new modification value of strut	Characteristic picture of clashes between strut and beam
1	First characteristic	$M = -(((Hc - dt) \times 2)) + T$ <p>Where:</p> <ul style="list-style-type: none"> <li>• M is a modification value</li> <li>• Hc is a height of beam</li> <li>• dt is an embedded thickness value of clash in horizontal direction</li> <li>• T is an allowance tolerance number</li> </ul>	
2	Second characteristic	$M = ((Hc \times 2) + T)$ <p>Where:</p> <ul style="list-style-type: none"> <li>• M is a modification value</li> <li>• Hc is a height of beam</li> <li>• dt is an embedded thickness value of clash in horizontal direction</li> <li>• T is an allowance tolerance number</li> </ul>	

Next, after the clash detection process, the clash results were analysed and saved in the spreadsheet. At the intersection position of clash, the system was designed to analyse the embedded thickness value in the transverse and horizontal directions. In each direction, it had the information of the detected element from the clash

problem. This information consisted of clash number, element type, element ID, strut type, strut ID, Point. X, Point. Y, Point. Z, element dimension and embedded thickness value. Table 9 shows an example of clash information between strut and the other three elements of building structures in the spreadsheet.

Table 9. An example of clashes between strut and three elements of building structures in the spreadsheet.

An example of clashes between strut and column structure													
	A	B	C	D	E	F	G	H	I	J	K	L	M
	Name	Building element	ID	Strut Type	ID	Point X(H)	Point Y(H)	Point Z(H)	Embedded thickness of clashpoint	Column (Width)	Column (Length)	Tolerance	Combined thickness of clash Point
1													
2	Clash0	Family Type: 500 x 900mm.	560922	Family Type: 400 x 400.	849333	23032.5	2398.8	-6500	250	500	900	300	700
3	Clash1	Family Type: 500 x 900mm.	560924	Family Type: 400 x 400.	849378	27331.9	2400.0	-6500	250	500	900	300	700
4	Clash2	Family Type: 500 x 900mm.	560929	Family Type: 400 x 400.	849448	57031.8	2400.0	-6500	250	500	900	300	700
5	Clash3	Family Type: 500 x 900mm.	560930	Family Type: 400 x 400.	849327	52031.9	2400.0	-6500	250	500	900	300	700
6	Clash4	Family Type: 500 x 900mm.	560931	Family Type: 400 x 400.	849372	43831.9	2400.0	-6500	250	500	900	300	700
7	Clash5	Family Type: 500 x 900mm.	560932	Family Type: 400 x 400.	849442	39631.9	2400.0	-6500	250	500	900	300	700
8	Clash6	Family Type: 500 x 900mm.	560933	Family Type: 400 x 400.	849327	35831.9	2400.0	-6500	250	500	900	300	700
9	Clash7	Family Type: CB2 0.4*1.0.	316465	Family Type: 400 x 400.	849372	-12745.0	-12692.5	-6900	250	400	1000	300	-600
10	Clash8	Family Type: CB2 0.4*1.0.	316467	Family Type: 400 x 400.	849442	-12745.0	-7692.5	-6900	250	400	1000	300	-600
11	Clash9	Family Type: CB2 0.4*1.0.	316469	Family Type: 400 x 400.	849298	-12745.0	-2392.5	-6900	356	400	1000	300	-388
12	Clash10	Family Type: CB2 0.4*1.0.	316471	Family Type: 400 x 400.	849327	-12745.0	5307.5	-6900	250	400	1000	300	-600
13	Clash11	Family Type: CB2 0.4*1.0.	316473	Family Type: 400 x 400.	849372	-12745.0	13007.5	-6900	250	400	1000	300	-600
14	Clash12	Family Type: CB2 0.4*1.0.	316485	Family Type: 400 x 400.	849397	-12745.0	-15392.5	-6900	356	400	1000	300	-388
15	Clash13	Family Type: CB2 0.4*1.0.	316358	Family Type: 400 x 400.	849413	11255.0	-2392.5	-6900	356	400	1000	300	-388
16	Clash14	Family Type: CB2 0.4*1.0.	316364	Family Type: 400 x 400.	849442	11255.0	-7692.5	-6900	250	400	1000	300	-600
17	Clash15	Family Type: CB2 0.4*1.0.	316376	Family Type: 400 x 400.	849298	11255.0	5307.5	-6900	358	400	1000	300	-384

Table 9. An example of clashes between strut and the other three elements of building structures in the spreadsheet (Continue).

An example of clashes between strut and wall structure												
	A	B	C	D	E	F	G	H	I	J	K	L
	Name	Building element	ID	Strut Type	ID	Point.X(V)	Point.Y(V)	Point.Z(V)	Embedded thickness of clash point	Wall (Width)	Tolerance	Combined thickness of clash point
1												
2	Clash0	W6 Brick mon- 200mm Brick	649797	Family Type: 400 x 400	849321	-12620.0	-13442.5	-6900	150	100	300	600
3	Clash1	W6 Brick mon- 200mm Brick	649799	Family Type: 400 x 400	849309	-12620.0	-15392.5	-6900	150	100	300	600
4	Clash2	W4 Brick mon- 100mm Brick	649814	Family Type: 400 x 400	849309	11480.0	10087.5	-6900	150	100	300	600
5	Clash3	W1 Light Weight - 100mm Masonry	649768	Family Type: 400 x 400	849309	-12620.0	-13467.5	-4200	150	100	300	600
6	Clash4	W6 Brick mon- 200mm Brick	649969	Family Type: 400 x 400	849366	-12570.0	-14667.5	-4200	150	100	300	600
7	Clash5	W4 Brick mon- 100mm Brick	649975	Family Type: 400 x 400	849436	11480.0	10087.5	-4200	150	100	300	600
8	Clash6	W1 Light Weight - 100mm Masonry	649832	Family Type: 400 x 400	849354	-12620.0	-13467.5	-1000	150	100	300	600
9	Clash7	W6 Brick mon- 200mm Brick	649969	Family Type: 400 x 400	849424	-12620.0	-15392.5	-1000	150	100	300	600
10	Clash8	W6 Brick mon- 200mm Brick	649974	Family Type: 400 x 400	849424	-12570.0	-14667.5	-1000	150	100	300	600
11	Clash9	W4 Brick mon- 100mm Brick	649975	Family Type: 400 x 400	849354	11480.0	10050.0	-1000	150	100	300	600
12	Clash10	W1 Light Weight - 100mm Masonry	649941	Family Type: 400 x 400	849424	-12620.0	-15192.5	-7100	150	100	300	600

An example of clashes between strut and beam structure													
	A	B	C	D	E	F	G	H	I	J	K	L	M
	Name	Building element	ID	Strut Type	ID	Point X(V)	Point Y(V)	Point Z(V)	Embedded thickness of clash point	Beam (Width)	Beam (Height)	Tolerance	Combined thickness of clash point
1													
2	Clash0	Family Type: BST2 0.25x0.5	316524	Family Type: 400 x 400	849296	23356.0	-7592.5	-4200	776	600	1000	300	-748
3	Clash1	Family Type: BBP1 1.2x0.4	316790	Family Type: 400 x 400	849315	23356.3	2505.1	-4200	776	600	1000	300	-748
4	Clash2	Family Type: BBP1 1.2x0.4	316547	Family Type: 400 x 400	849302	35506.0	-15382.5	-4200	450	600	1000	300	1200
5	Clash3	Family Type: BBP1 1.2x0.4	316549	Family Type: 400 x 400	849315	35506.0	-7592.5	-4200	450	600	1000	300	1200
6	Clash4	Family Type: BP1 0.35x0.7	316544	Family Type: 400 x 400	849302	35506.0	2507.5	-4200	776	600	1000	300	-748
7	Clash5	Family Type: BBP1 1.2x0.4	316547	Family Type: 400 x 400	849315	35506.0	10357.5	-4200	776	350	900	300	-548
8	Clash6	Family Type: BP1 0.35x0.7	316553	Family Type: 400 x 400	849315	35506.0	13132.5	-4200	776	350	900	300	-548
9	Clash7	Family Type: BP1 0.35x0.7	316764	Family Type: 400 x 400	849316	23032.5	2398.8	-3800	776	500	900	300	-548
10	Clash8	Family Type: BP1 0.35x0.7	316766	Family Type: 400 x 400	849360	27331.9	2400.0	-3800	776	500	900	300	-548
11	Clash9	Family Type: BP1 0.35x0.7	316784	Family Type: 400 x 400	849361	57031.8	2400.0	-3800	776	500	900	300	-548
12	Clash10	Family Type: BP1 0.35x0.7	316786	Family Type: 400 x 400	849296	52031.9	2400.0	-3800	776	600	1000	300	-748
13	Clash11	Family Type: BP1 0.35x0.7	316961	Family Type: 400 x 400	849364	43831.9	2400.0	-3800	25	500	900	300	350

To automatically modify the strut location, it required to separately relocate the strut in transverse and horizontal directions. In the transverse direction, the input information needed to obtain from the 3D model and spreadsheet. The 3D strut and strut ID were extracted from the 3D model whereas the strut ID and embedded thickness value of detected wall and column. Then, the rule-based approach was applied to (1) match between strut ID of detected wall and strut ID of detected column, (2) determine whether embedded thickness value of detected wall was bigger than embedded thickness value

of detected column, (3) match between strut ID from 3Dmodel and strut ID of detected column and wall. Finally, the output of module could specify the results of (1) the embedded thickness value of detected column and wall, (2) new embedded thickness value, (3) new coordinate value (X, Y, Z), and (4) new strut location in transverse direction. Figure 9 shows the flow chart information for modifying the strut location in the transverse direction. Moreover, the strut relocation in horizontal direction also followed the same process in the transverse direction.

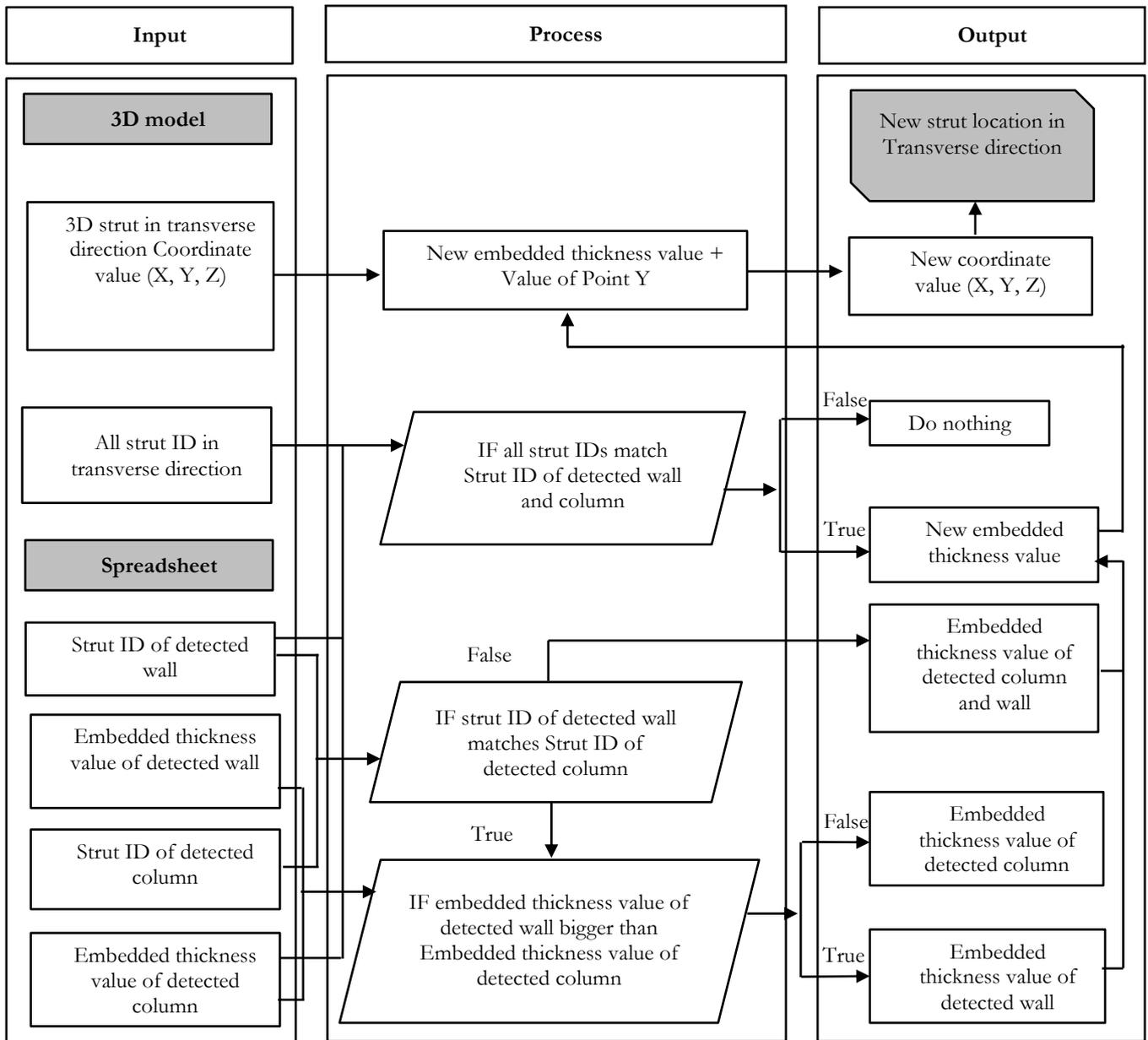


Fig. 9. Flow chart information for modifying the strut location in the transverse direction.

## 5. System Testing

### 5.1. Project Description

To validate the system development, a case study was a newly constructed project of office building in Bangkok, Thailand. There were two buildings in this project and the building height is 22.9 m from the ground floor to the seventh floor of both buildings. Moreover, each building

had three floors in the basement and a total area of both basements was approximately 4,911.2 m<sup>2</sup>. Figure 10 shows the basement level of the office building project in 3D model. These basements were used to serve as car parking lots. In this excavation, the width number was around 31.5 m, the length number was around 77.7 m and the depth number was around 14.6 m. Moreover, the basement structure, which consisted of concrete piles, footings, columns, slabs, walls, and beams, was developed in the 3D model by using Autodesk Revit Software.

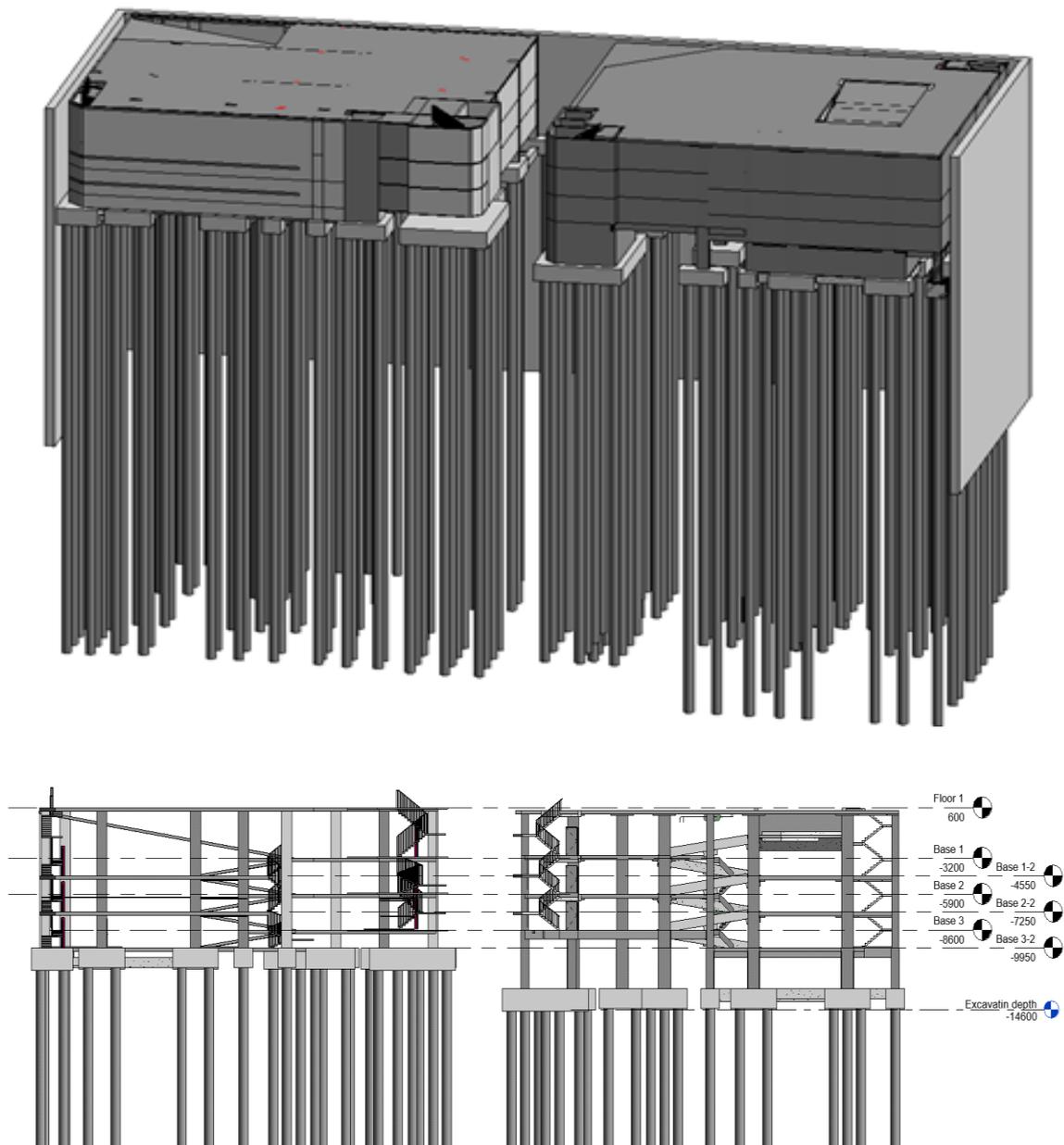


Fig. 10. A 3D basement model of the office building project.

## 5.2. Results of A Case Study Testing

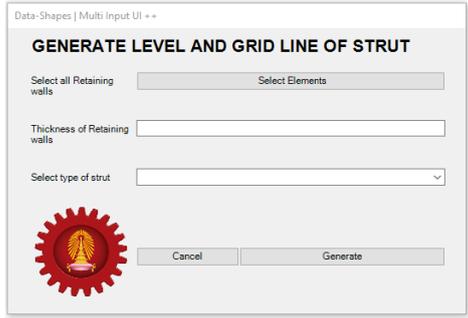
In this study, the 3D basement model of the office building project was applied to test the system development. The system development was applied with the 3D basement model for serving the main functions of strut arrangement including (1) strut generation, (2) clash detection of strut, (3) automated strut modification. The input and result of each function were described in the following section.

### 5.2.1. Step 1: Strut generation

#### 5.2.1.1. Input data process

The input information, which was required in the structural strut generation, was the boundary line of retaining wall and thickness of retaining wall. The boundary line of the retaining wall was obtained by the model element selection from the 3D model. Then, the thickness of the retaining wall was also provided with the actual number of retaining wall thickness. Third, the type of strut was selected among three strut sizes including 300 x 300 mm, 350 x 350 mm, and 400 x 400 mm. Table 10 shows the input information and interface of strut generation. Moreover, the space number between strut and strut is around 6000 mm. Last, after inputting the information requirement, the dynamo runs the calculation process for determining the numbers of strut level and strut quantities in transverse and horizontal directions.

Table 10. Input information and interface of strut generation.

Input information	Dynamo input interface
Boundary line of retaining wall -> Select model elements Thickness of retaining wall -> 800 mm.	

5.2.1.2. Result of structural strut generation

Due to the depth of excavation around 14.6m, the concrete pile was selected as the material of retaining wall. Moreover, the thickness of the retaining wall was around 800mm. After applying the input information requirement, the Autodesk Dynamo Software was run by

combining with the algorithm of strut generation. As a result, the system could determine 4 strut levels and strut quantities consisted of 24 horizontal struts and 56 transverse struts. This result was also illustrated in the 3D model. Table 11 shows the result of strut generation with a case study. Figure 11 shows an initial structural strut generation of a case study in the 3D model.

Table 11. Result of strut generation with a case study.

Steps of strut generation	Results	Strut levels
1. Numbers of strut level 2. Strut quantities in transverse direction 3. Strut quantities in horizontal direction 4. Depth of excavation	4 levels $14 \times 4 = 56$ struts $6 \times 4 = 24$ struts -14.6mm	First level = -800mm Second level = -4000mm Third level = -6700mm Fourth level = -9300mm



Fig. 11. An initial structural strut generation of a case study

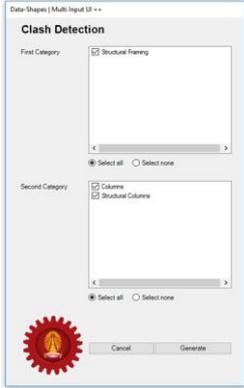
## 5.2.2. Step 2: Clash detection of strut

## 5.2.2.1. Input data process of clash detection

The input information, which was required in the clash detection, was the selection between strut and permanent building structures. In the permanent building structures, it considered only column, wall, and beam because the clash detection between (1) strut and

foundation, (2) strut and slab were avoided by the strut generation stage. The selection between strut and permanent building structures was applied case by case which had (1) clash detection between strut and column, (2) clash detection between strut and wall and (3) clash detection between strut and beam. Each clash detection case shown the input requirement and interface in Table 12.

Table 12. Input requirement and interface of each clash detection case.

Input information	Dynamo interface
Clash detection between strut and column	
<p>Select model elements -&gt; Select strut from the 3D model Select model elements -&gt; Select column from the 3D model</p>	
Clash detection between strut and wall	
<p>Select model elements -&gt; Select strut from the 3D model Select model elements -&gt; Select wall from the 3D model</p>	
Clash detection between strut and beam	
<p>Select model elements -&gt; Select strut from the 3D model Select model elements -&gt; Select beam from the 3D model</p>	

### 5.2.2.2. Result of clash detection

In the clash detection, the permanent building structures, which were included column, wall, and beam, were selected to check the clash problems with the structural strut. As a result, the clash numbers, clash points, and embedded thickness values of clashes were identified and extracted in the spreadsheet. Moreover, the detected struts and permanent building structures were shown in the 3D model. Table 13 shows the results of clash numbers between struts and permanent building

structures. Figure 12 shows the results of detected struts and permanent building structures in the 3D model. Moreover, when it had a clash between strut and building elements like column, wall or beam, it shown the color of clashed element. For example, when it had a clash, strut was changed to green color, beam was changed to blue color, wall was changed to red color and column was changed to orange color. On the other hand, when the element was not clashed, it was not changed to any color in 3D view.

Table 13. Results of clash numbers between struts and permanent building structures.

Clash detection	Clash numbers
Clash detection between strut and column	95 clashes
Clash detection between strut and wall	11 clashes
Clash detection between strut and beam	31 clashes

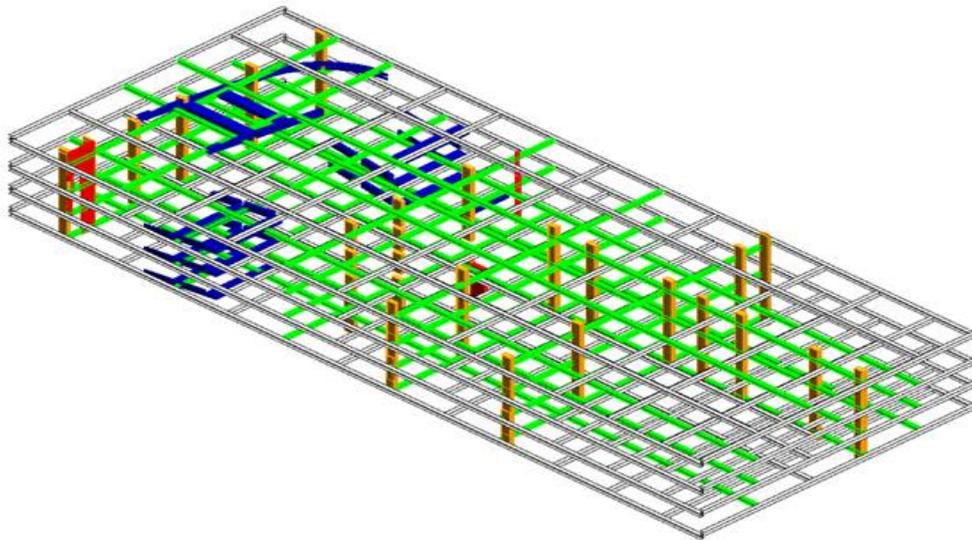


Fig. 12. Detected struts and permanent building structures in the 3D model.

### 5.2.3. Step 3: An automated strut modification

#### 5.2.3.1. Input data process of an automated strut modification

The input information, which only considered the strut element, was applied by selecting the strut in the 3D model. The automated strut modification could solve the clash problems between strut and permanent building structures. Table 14 shows the input requirement and interface of an automated strut modification.

Table 14. Input requirement and interface of an automated strut modification.

Input information	Dynamo interface
Select all struts in the 3D model	

### 5.2.3.2. Result of an automated strut modification

After applying the input requirement of strut modification, The Autodesk Dynamo Software extracted the data from the spreadsheet. In this data, the re-located

value was the sum between the embedded thickness value of clash and tolerance value. The strut was moved by the re-located value. As a result, the final structural strut was obtained without the clash problems and shown in Fig. 13.

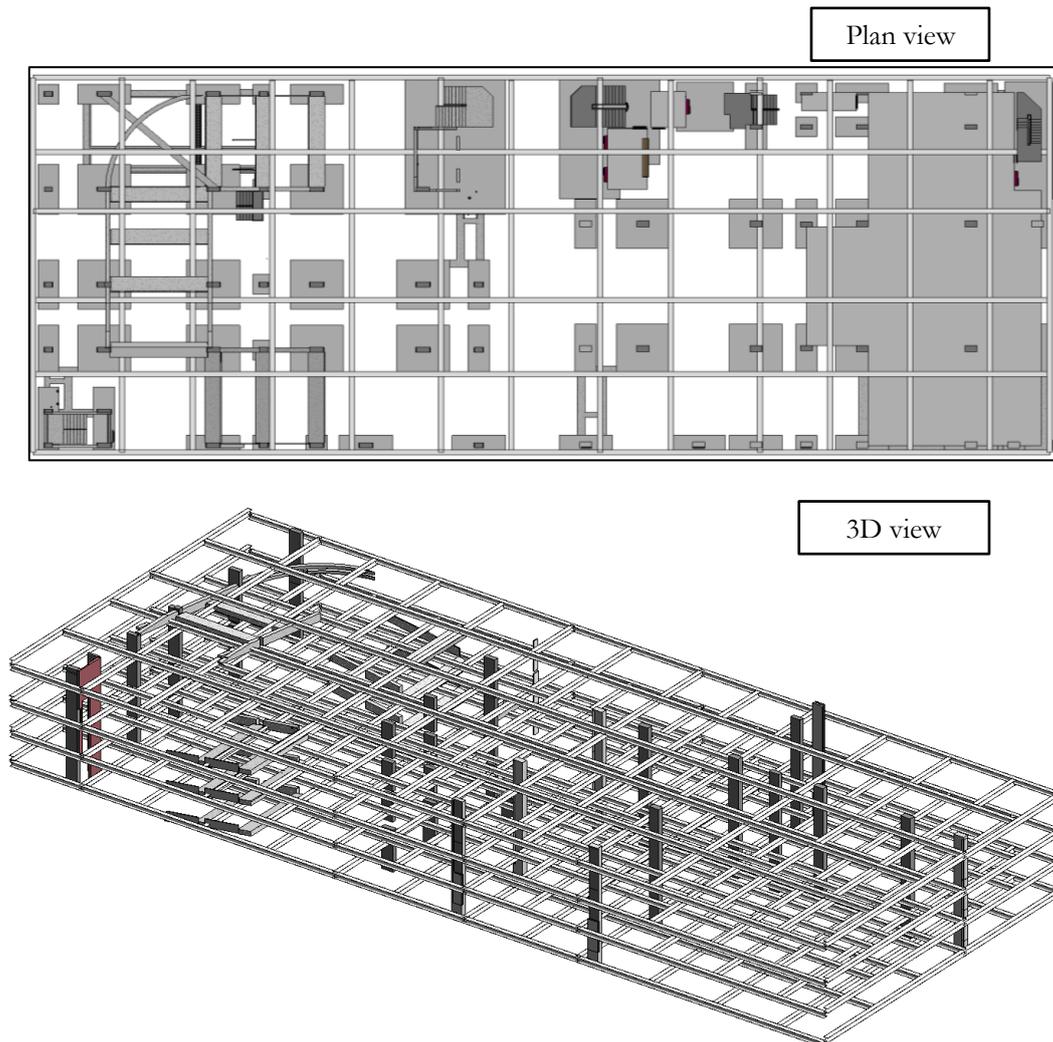


Fig. 13. A final structural strut in the 3D model.

## 6. Discussion of System Development

The finding results of this system development have significantly supported the strut arrangement. Each finding is clearly explained in the next section.

First, the system in this study could be the first system that was developed to support the strut arrangement. After developing and applying the automated system for supporting the strut arrangement, its capability could effectively serve three functions of the strut arrangement. First, in the structural generation function, when the depth of excavation is smaller or bigger than 9 m, the strut generation could determine the numbers of strut level. Moreover, the strut quantities in both transverse and horizontal directions were determined with any space number given. As a result, the initial structural strut could generate into a 3D model. Next, in clash detection function of strut, the initial structural strut was run and detected the clash problems with the 3D basement model.

Then, the detection result could provide as either display in the 3D model or save in the spreadsheet. Thus, the engineer could see the problems in the 3D model and find the modification information in the spreadsheet. Last, the strut modification function could automatically relocate the strut location for solving the clash problems. Then, the final structural strut could obtain without the clash problem in the 3D model. In short, each function of this system development was properly worked to support the strut arrangement.

In this study, the structural strut generation function was developed without considering the design work of the strutting system. Due to different ground layers in deep excavation, they could have different lateral forces and lead to determine different numbers of strut sizes. However, the strut generation in this study used the fixing number of strut size which has three main sizes of strut namely 300 x 300 mm, 350 x 350 mm, and 400 x 400 mm. Thus, the engineers could separately use each function of

the strut arrangement. They could skip the structural strut generation function and directly used the clash detection and strut modification functions.

Next, it is related to the visualization aspect. The system is basically depended on the 3D model, so it could change the current practice of strut arrangement work. In the current practice, the strut arrangement is usually based on two-dimensional drawings, this practice still is limited to clearly understand the clash problems between strut and permanent building structures. Thus, this system was developed by visualizing the problems in the 3D model. Due to the information of strut arrangement in real practice, the algorithm of clash detection was designed to distinguish between occurrence and non-occurrence of clash. Then, this algorithm developed with BIM technology for displaying all clash occurrences on each strut at the same time. As a result, all clash problems are successfully visualized on each strut. Therefore, the detective function of this system was useful to display the detected clashes in the 3D model.

Finally, this system also has shortened the time of the strut arrangement. Based on the current practice of strut arrangement, this work is time-consuming and error-prone. This automated system was developed by using the computer ability with the information of strut arrangement to distinguish between occurrence and non-occurrence of clash. Thus, it could reduce the time for detecting and solving the clash problems in the strut arrangement. Therefore, the result of strut arrangement work is shortly obtained rather than the current practice.

## 7. Conclusion

In the current practice, the strut arrangement is usually based on two-dimensional drawing, this practice is still limited to visualize the clash between strut and permanent building structures. After some available software tools are developed to overcome this problem, they could display the clash problem one by one. However, it is difficult for the engineer to oversee all the clash occurrences on each strut at the same time. Moreover, the available software is still unable to distinguish between occurrence and non-occurrence of clash based on real practice. Thus, this study adopts Building Information Modelling (BIM) to develop a system for automatically detecting and solving clashes of the structural strut. In the system development, the algorithms were designed from the information of strut arrangement. This information was obtained by the experts' interviews in the deep excavation of building projects and presented in another paper. After the algorithms were developed to serve three functions of strut arrangement including strut generation, strut detection, and strut modification, the system was created by using Autodesk Revit and Dynamo Software and Microsoft Excel Software. Next, system development was tested to understand its efficiency with a case study. As a result, this system could effectively identify the clash problems between strut and permanent building

structures like column, wall, and beam. Moreover, this system could solve the clash problems by automatically relocating the strut to another location. Last, this system also provides a clear picture by visualizing the clash problems between strut and permanent building structures in the 3D model.

There are two main contributions to this system development. First of all, due to the significant point of this system development, the strut arrangement work could strengthen the strut design. Initially, the result of this strut arrangement provides the numbers of strut level, strut spaces, and quantities in both directions. Moreover, the system could be applied to check the clash problems too. Then, the designer could use this information to confirm the specification of strut design. Thus, the designer could ensure the structural strut in design and construction work. Moreover, the designer also could use the final structural strut to take off the strut quantity and estimate the cost.

Next, this system could change the process of structural strut development. The designer usually needs to design the structural strut by determining the element specification like types of retaining wall, numbers of strut level, strut size and space. Then, the designer arranges and checks the clash problems between strut and permanent building structures. However, this automated system could reverse the process of strut arrangement. The designer primarily has to create the basement structure in the 3D model and apply this system development to arrange the structural strut at the first step. After this strut arrangement work is complete, it can send for determining the strut specification later. Thus, it could support the designer's work. Moreover, some subcontractors could use the 3D strut model to estimate the primary cost of the structural strut. In short, the finding result of the system development has significantly changed the current practice of strut arrangement at the primary stage of deep excavation work.

Although the automated system for supporting the strut arrangement is developed successfully, it still has a few limitations that could influence this system. First, based on system development, it could allow the user to see the clash problems between strut and permanent building structures at the pre-construction stage of the project. However, in the structural generation function, it was developed by using the reference value of strut level and space from many previous projects in the deep excavation. This study still encourages the user to bring the structural strut, which has applied with this system, to check with the design work of the strutting system. So it can confirm with the specification of the structural strut from the design work. This practice is considered as a double-check for ensuring the stability of the strutting system. Next, the current practice uses the AutoCAD Software to develop the structural strut in two-dimensional drawing. If designer or subcontractor would like to use this system development, he or she needs other software like Autodesk Revit and Dynamo Software. Thus, it may have an extra cost for changing the

AutoCAD to Autodesk Revit and Dynamo Software. Finally, the future study should focus on other types of constructible problems in the strut arrangements like checking tolerance problem between strut and permanent building structures and spacing problem between strut and strut.

## Acknowledgments

This paper was financially supported by the 90th Anniversary Fund of Chulalongkorn University (Ratchadaphiseksomphot Endowment Fund) and the ASEAN scholarship program.

## References

- [1] S. H. Chao, N. B. Karki, and D. R. Sahoo, "Seismic behavior of steel buildings with hybrid braced frames," *Journal of Structural Engineering*, vol. 139, no. 6, pp. 1019-1032, 2013.
- [2] Y. Haihong and H. Chengpo, "The strut arrangement and monitoring analysis of deep foundation pits in silty sand," in *Advances in Civil Engineering and Building Materials*. London: Taylor and Francis, 2013,, pp. 769-772.
- [3] I. Askew, "Deep excavations—Industry challenges," in *APEC Seminar on The State-of-the-Practice of Deep Excavation Works*, Hong Kong, Geotechnical Division, The Hong Kong Institution of Engineers, 2011.
- [4] H. M. Lin and F. C. Hadiprino, "Problems in deep foundation construction in Taiwan," *Journal of Performance of Constructed Facilities*, vol. 4, no. 4, pp. 259-270, 1991.
- [5] Y. Alostaz, A. Hagh, and J. Pecora, "Design and construction of a support of excavation system for a deep cut-and-cover tunnel in Downtown Boston," in *Proc. of the Fifth International Conference on Case Histories in Geotechnical Engineering*, Massachusetts, 2004, vol. 26.
- [6] W. A. Marr and M. Hawkes, "Displacement-based design for deep excavations," *Geotechnical Special Publication*, vol. 384, no. 208, pp. 82-100, 2010.
- [7] L. Y. Ding, B. T. Zhong, S. Wu, and H. B. Luo, "Construction risk knowledge management in BIM using ontology and semantic web technology," *Safety Science*, vol. 87, pp. 202-213, 2016.
- [8] H. Luo and P. Gong, "A BIM-based code compliance checking process of deep foundation construction plans," *Journal of Intelligent and Robotic Systems: Theory and Applications*, vol. 79, pp. 549-576, 2015.
- [9] J. Gondar and A. Pinto, "Case study: BIM and geotechnical project in urban area – Infinity Tower," in *Proceedings of China-Europe Conference on Geotechnical Engineering*, 2018, pp. 923-926.
- [10] M. Choi, G. Lee, and H. Kim, "A framework for evaluating deep excavation alternatives in building construction," in *International Conference on Computing in Civil and Building Engineering (ICCCBE)*, 2008, pp.1-6.
- [11] H. Luo and P. Gong, "A BIM-based code compliance checking process of deep foundation construction plans," *Journal of Intelligent and Robotic Systems: Theory and Applications*, vol. 79, pp. 549-576, 2015.
- [12] Y. Tian and H. Yue, "Application research of BIM technology in computer aided design of building foundation," *Chemical Engineering Transactions*, vol. 51, pp. 1427-432, 2016.
- [13] M. M. Singh, A. Sawhney, and A. Borrmann, "Integrating rules of modular coordination to improve model authoring in BIM," *International Journal of Construction Management*, vol. 19, no.1, pp. 15-31, 2019, doi:10.1080/15623599.2017.1358077.
- [14] S. R. Lu, I. C. Wu, and B. C. B. Hsiung, "Applying BIM in environmental impact assessment for urban deep excavation projects," in *Proceedings of the 29th ISARC*, Eindhoven, The Netherlands, 2012.
- [15] I. C. Wu, S. R. Lu, and B. C. Hsiung, "A BIM-based monitoring system for urban deep excavation projects," *Visualization in Engineering*, vol. 3, pp. 1-11, 2015.
- [16] J. Lin, J. Zhang, Q. Wen, and F. Wang, "Leveraging BIM in settlement monitoring and impact management for subway excavation," presented at *Conferences – CIB W78*, Eindhoven, The Netherlands, 2015.
- [17] D. Szwarkowski and E. Pilecka, "BIM technology in geotechnical engineering in terms of impact high building "Mogilska Tower" in Cracow of existing building development," *Technical Sciences*, vol. 20, pp. 297-309, 2017.
- [18] C. Tanaka and C. I. W. Hong, "Application of BIM in geotechnics—Case study on a deep excavation project," in *Proceedings of the 6th International Young Geotechnical Engineers' Conference*, Seoul, Korea, 2017.



**Pisal Nov** was born in October 27, 1987 in Phnom Penh, Cambodia. He received Bachelor Degree from two Universities. One university, he graduated Bachelor Degree of Civil Engineering at Norton University in 2010. Another one, he graduated Bachelor Degree of Professional Communication (PC) in English at Institute of Foreign Languages (IFL) in 2010. In 2011 and 2013, he was awarded by ASEAN country scholarship to continue the M.S. and Ph.D. Degree in field of Construction Engineering and Management, Chulalongkorn University, Thailand.

Since 2019, he has been a lecturer with Civil Engineering Department, Paragon International University. His research interests include construction project planning, construction contract and construction technology like building information modeling (BIM).

Dr. Pisal Nov is a member of Engineering Board Cambodia.



**Vachara Peansupap** received the B.S. in Civil Engineering from King Mongkut's Institute of Technology Ladkrabang, Thailand in 1994. In 1997, he graduated Master Degree in Construction Engineering and Management from Chulalongkorn University, Thailand. In 2005, he received the Ph.D. Degree in Built Environment from RMIT University, Australia.

Since 1998, he has been an Associate Professor with the Civil Engineering Department, Chulalongkorn University, Thailand. His research interests include construction innovation, knowledge management, construction automation, building information modeling (BIM), lean construction, quality management.

Assoc. Prof. Dr. Vachara Peansupap is a member of Council of Engineers.



**Tanit Tongthong** received the B.Eng. in Civil Engineering from Chulalongkorn University, Thailand in 1986. In 1989, he graduated M.S. in Civil Engineering from University of Missouri, United States. In 1993, he received the Ph.D. Degree in Civil Engineering from University of Maryland, United States.

Since 1994, he has been an Associate Professor with the Civil Engineering Department, Chulalongkorn University, Thailand. His research interests include construction technology and innovation, construction management, building information modeling (BIM), quality management in construction.

Assoc. Prof. Dr. Tanit Tongthong is a Director Accreditation Program (DAP 11/2004) and an Advanced Audit Committee Program (AACP 29/2018)