MODELLING EVACUATION STRATEGIES UNDER DYNAMIC CONDITIONS DUE TO OBSTACLE LOCATIONS BASED ON A SEMANTIC 3D BUILDING MODELS

Shreya, K.S Rajan

Lab for Spatial Informatics, IIIT Hyderabad, Hyderabad, Telangana-500032

KEY WORDS: Building Evacuation, 3D floor plan, Evacuation Strategies, IndoorGML, IFC, Obstacles, Path Planning.

ABSTRACT:

The evacuation path from inside a building to safe point outside becomes highly unpredictable due to changes in the local geometry or presence of obstacles during a disaster like a fire. During emergencies, Evacuees need appropriate information and hence prediction of an unobstructed path, as it emerges, needs to be computed well. Understanding the exits with its allowable people flow rate; the type - door or alternative exit such as windows, balconies, etc.; and its role as a node in the graph network is important to ensure safe and timely evacuation from a building. The study here evaluates how obstacles present in the evacuation route affect the removal of the last person. These obstacles, such as furniture, decrease the flow rate at which evacuees can escape. A subspace model is proposed for *geometric spaces or carpet areas* containing obstacles and is used to compute the shortest obstacle-free paths. The occupancy is considered within the subspaces containing obstacles. The proposed method clearly shows that a graph-based path generation using a subspace model improves the computation time, can be dynamically adapted, and can be scalable across geometric spaces. The results clearly show the impact of the obstacles, with a 2x to 6x rise when compared to obstacle-free scenarios.

1. INTRODUCTION

The complex nature of evacuation path planning in an emergency requires a range of information, both static and dynamic. Evacuees are also concerned about their escape in spaces with obstacles. A geospatial model of a 3D floor plan consisting of not just the space geometry but also the location of various objects along with their dimensions can help to visualize and understand how the graph needs to be generated for an obstacle-free path. Semantic, geometric, and topologic transformation is an important aspect in order to derive the appropriate navigation structures, (Khan et al, 2014) describes the multistep transformation process and sub-spacing approach to get the indoor Geography markup language (GML) LoD4 dataset automatically from the existing semantic 3D building model according to Industry Foundation Classes (IFC). The 3D model is being used here for not just obstacles but to basically state that one cannot cross over the obstacles if it is beyond the limits of the person, and will have to go around that obstacles, leading to increased path travel time. That will lead to the need to generate new networks and paths as appropriate. IndoorGML Level of Detail (LoD) 4 is important for the requirement of indoor spatial information. It works as an anchor for the connection between indoor and outdoor environments. Interoperability between Building Information Modeling (BIM) and Geographic information systems (GIS) has a strong benefit for different use cases such as path planning for evacuation, urban planning, etc. The conversion of BIM to 3D by integrating semantic data enhances the process of urban planning. Most of the research efforts have been made for the integration of semantic models such as IFC and CityGML using the unidirectional approach for data conversion, (El-Mekawy et al, 2012) investigated the potential of the conversion between IFC and CityGML and concluded that the integration of two different systems is somewhat difficult to develop for many reasons. The review of the relevant research papers is discussed in (Zhu et al, 2018). According to this GIS and BIM will operate as an independent system but the priority is to get effective interoperability data. IFC and CityGML are most studied and accepted for the exchange of building data in the format such as shapefile known as multipatch. An algorithm for free multi-floor indoor space extraction from a 3D building model is introduced. As most of them had a conversion for IFC to CityGML but nobody clearly talked about the process of conversion therefore this paper presents the conversion process from IFC to Shapefile using a Feature manipulation engine (FME). Since the establishment of the building, the natural or unnatural damages to the buildings, and changes done over time can lead to different space utilization. So, (Liu at el, 2011) a door-to-door evacuation model is proposed using geometric and semantic information and providing evacuation instructions to the people considering local environmental factors in the corridors and common spaces whereas (Lui at el, 2016) gives the best-known path from multiple exits in the building. Indoor navigation based on an ontology of indoor space is the foundation for evacuation planning as well as navigation for evacuees, (Ma et el, 2017) introduced indoor navigation based on a two-level routing strategy and tested it with a variety of complex cases. This work has also been extended to 3D.

1.1 Objective

This paper introduces a subspace model, at whose centroid the people within the subspace congregate, that are then used to generate an unhindered path around the objects to the room door and further on to the building exit. To assess the correctness of the approach, the paper presents a comparison with an earlier work (Shreya and Rajan, 2023) on an unobstructed path in a free space, against the proposed approach with obstacles in both normal and adverse conditions. As the earlier studies do not consider the path extraction through the obstacles, the changes in the indoor environment, and the emergence of alternate available exits, it is difficult to assess the true impact of those methods on the evacuation of densely populated public and large buildings. This paper assumes that the 3D data includes obstacles and other key space features and hence does not cover the identification, mapping, or scaling of obstacles.

2. METHODOLOGY

2.1 Transformation of 3D building to CityGML LoD4

2.1.1 From IFC to Sketchup IFC stands for industrial foundation class, and it is an international standard for Architecture, Engineering, and Construction (AEC) representation. On the other hand, Sketchup is a content creator tool for 3D modeling Computer Aided Design (CAD). It is a user-friendly tool with a wide range of drawing and design applications. IFC information such as semantic and geometric data was imported to Sketchup pro application. Sketchup pro model was generated using a cartesian (local) coordinate system and a bounding box was generated for the 3D model using Thomas Thomassen's extensions such as DrawBoundingBox and `tt_Lib`[2]. 3D model tags were used for CityGML semantic elements. Abbreviation such as ws→ WallSurface, gs→GroundSurface, w→ Window, d→ Door, bi→ BuildingInstallation was used. Some other abbreviations are "ws-1_w-1" and "ws-1_d-1", which interprets as "windows on the wall surface 1" and "doors on the wall surface 1" respectively. Parent id (gml parent ID) and child element ID (gml ID) were separated with an underscore (_) and their own ids are separated with a dash (-).*Figure 1* shows the conversion from IFC to Sketchup with tags.

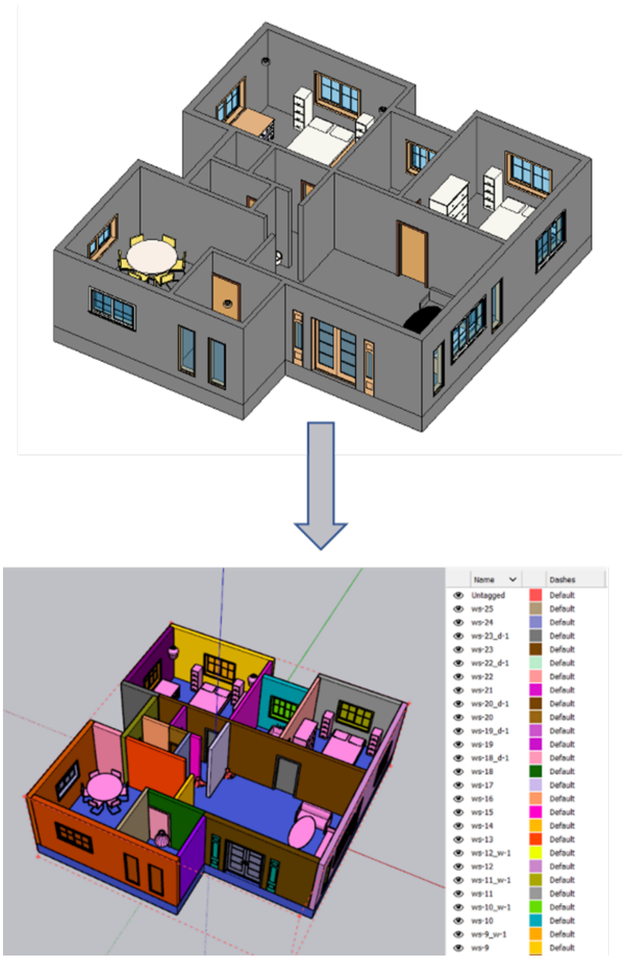

Figure 1. IFC to Sketchup conversion with tags.

2.1.2 From Sketchup to FME workbench Feature manipulation engine (FME) is a spatial extract, transform, and load (ETL) tool for data integration. It supports more than 400 different data formats. It allows users to develop workflow graphically for integration, automation, and translation of data. In this paper, the SketchUp (.skp) file serves as the input data, and it is imported into FME as a reader file. A workbench is created within FME, utilizing a variety of transformers to perform specific tasks. The generation of the FME workbench is elaborated in detail in *section 2.1.3* of this paper. This section outlines the specific transformers and their configurations used within the workbench to achieve the desired data transformation and integration.

2.1.3 From FME workbench to CityGML CityGML is known to comply with Open Geospatial Consortium (OGC) standards. CityGML is an open data model that is used for the exchange of 3D city models and also for storage. The unified Modelling Language (UML) object model defines GML. In CityGML, an object of the building can be defined through the Level Of Detail (LoD). There are a total of 5 LoD's - LoD0 shows the footprint of the building on the ground surface; LoD1 is a simple extrusion from the footprint that forms a block-like model known as the block model; in LoD2, the objects are given a form that resembles the real world with Building block models given roof surfaces; LoD3 captures the building key components such as doors, windows, stairs, pipes, lamps, etc.;and finally in LoD4 the models of the interior objects such as furniture and equipment elements are captured and stored. In this study, the FME workbench was created using different transformers such as GeometryPartExtractor, Deaggregator, Aggregator, CityGMLGeometrySetter, and CoordinatSystemSetter. GeometryPartExtractor, allows to extract a part of underlying geometry using a geometry query. In SketchUp, as mentioned objects were grouped and had multi-level aggregate geometry which does not fit well with the basics of building geometry of City GML, so Deaggregator is used to flatten all the levels and split it into its component of faces, whereas the aggregator generates the backup to single multi-surface. CityGML geometry was set using CityGMLGeometrySetter, such as LoD to multisurface and as it is a basic building geometry, so is selected as city object member. The coordinate system was reset to the SketchUp coordinate system using the CoordinateSystemSetter transformer. The workbench tree is shown in *Figure 2*. The conversion process generates the final GML output that can be viewed in any viewer for semantic data models. Here, FZK Viewer tool has been used to visualize the semantic data models as shown in *Figure 3*.

2.2 Conversion from CityGML LoD4 to 2D floor plan

Conversion of CityGML LoD 4 was implemented using the output writer file as shapefile output. This shapefile output was imported to QGIS with the help of the vector layer tool and the projection was set up to ensure the right orientation of the 2D floor plan. The 2D floor plan consists of six accessible room spaces, six internal doors and one main door (or building exit), and twelve windows among which six are blocked due to the presence of obstacles that make these windows not accessible or useful as an alternate exit. In this floor plan, there are some blocked spaces that are also not accessible. The 2D floor plan along with the furniture is shown in *Figure 4*.

2.3 Subspace formation

The subspace model is based on the corner points of an obstacle. Each subspace will have an unhindered line of sight to the other subspace which is further connected to the intermediate or final exit door. To get a clear line of sight to another

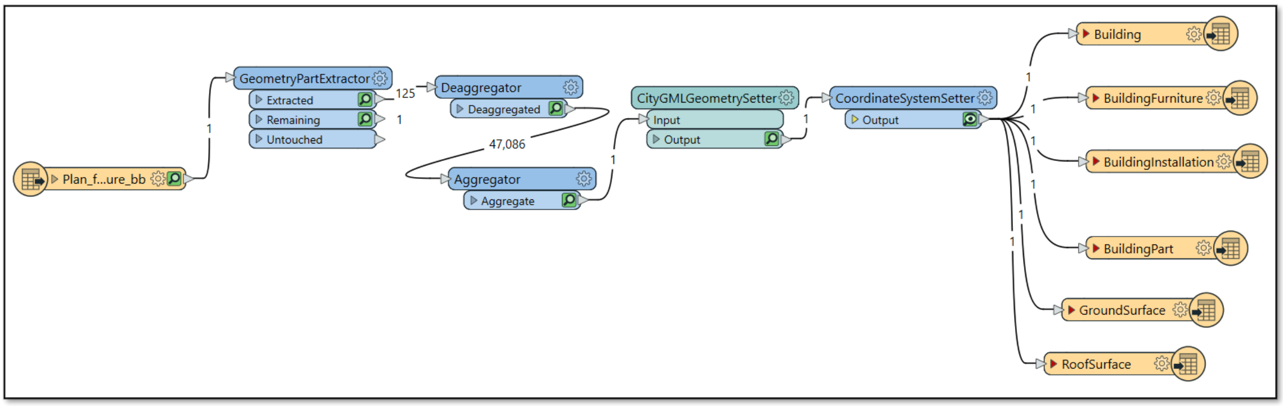

Figure 2. FME Workbench tree

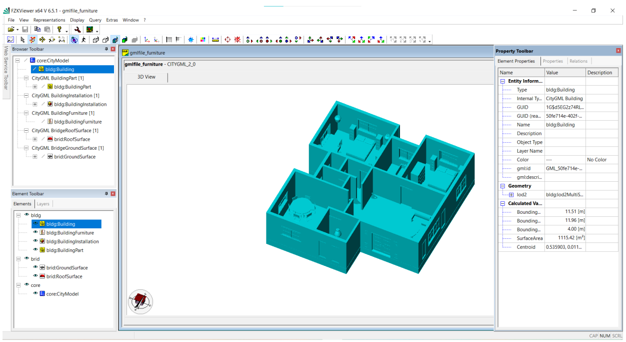

Figure 3. Visualization of the semantic data model

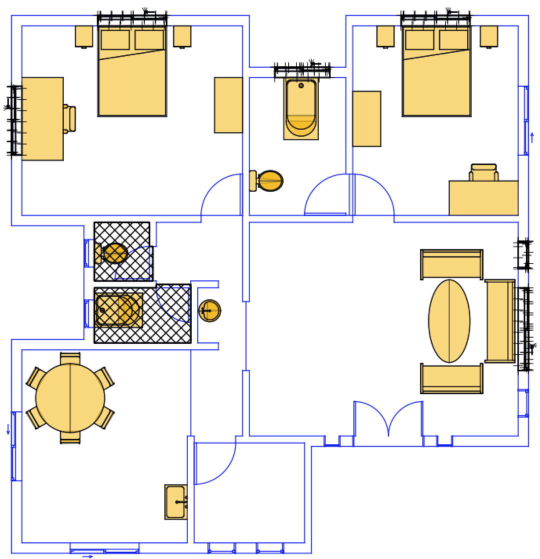

Figure 4. 2D floor plan with furniture in QGIS

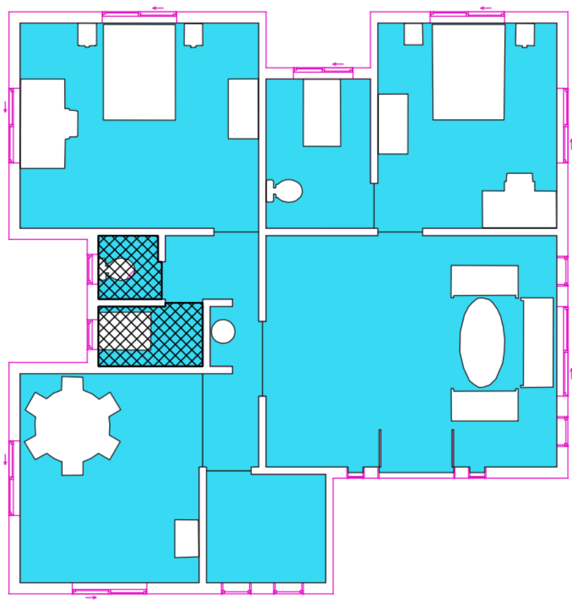

Figure 5. Extracted obstacles in the floor plan

subspace the intermediate node i.e.: the extra node (*Figure 6*) is generated. Post that these subspaces are used for all further network or path generation. The subspaces are shown in *Figure 6.*

2.4 Extraction of Obstacles

The generated 2D floor plan in *section 2.2* has the furniture elements that interrupt the path during the evacuation, and hence are considered as an obstacle. These obstacles were extracted as a polygon as in *Figure 5*. These extracted polygons were further used for the generation of subspaces.

2.4.1 Extraction of Nodes Subspace formation helps to extract the nodes at the centroid of these subspaces. The intermediate nodes were located at the centroid of the doors and windows that were accessible. *Figure 6* shows the subspaces and the corresponding nodes of the network. In cases where the direct line of sight between nodes of connected subspaces and/or exits was not possible, a few extra nodes were also computed based on the geometry.

2.4.2 Graph network Graph network is one of the most important key aspects of evacuation. This graph network was generated using the centroids extracted earlier. Coordinates were extracted in the attribute table of QGIS and were exported to .csv file format. Python, a high-level programming language was used to generate the network in QGIS. Several libraries were used such as “NumPy”, “pandas” and “matplotlib”. NumPy is used for simple calculation and mathematical operations on an array whereas pandas and matplotlib are used for data handling, exporting, and graph plotting respectively. The

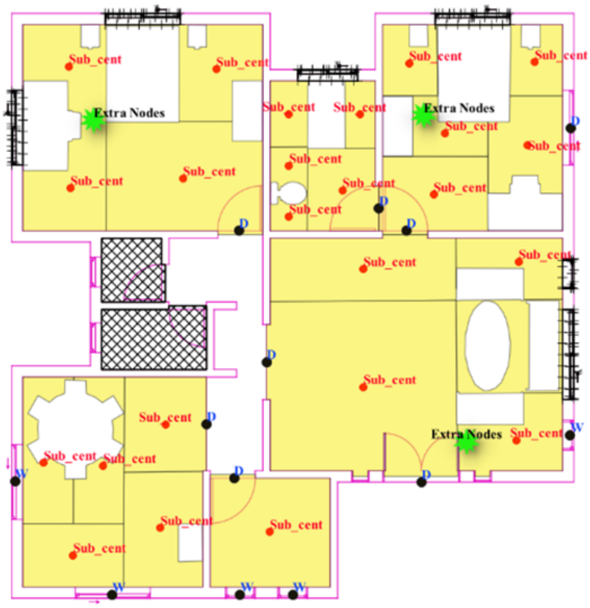

Figure 6. Centroids at subspaces, doors, and windows

.csv file was first imported and connection points for each and every node was defined as extracted from the csv file. These connection point provided the coordinates for generating a suitable graph. Further these were imported into QGIS, and was joined by "manage to join to other layers" with the point layer. The "XY to line" feature for simple lines based on starting and ending coordinates was used. The table data record contains the origins (X_0,Y_0), and destinations (X_d,Y_d). A coordinate reference system (CRS) was appropriately selected for the input coordinates. *Figure 7* shows the network generated using this approach.

Figure 7. Network generated using Python

2.5 Path for Evacuation

The network that is generated in the above section is important for calculating distances between nodes, considering factors like *distance-time-edge capacity*. To estimate the time, an average speed of 6 kmph per person is assumed, accounting for evacuation movement behavior. This allows for the computation of each occupant's travel time to the exit and the estimation of the last person's exit time. The algorithm proposed earlier, as an extension to (Shreya and Rajan, 2023), is employed within the PostgreSQL system to generate evacuation paths. By utilizing this algorithm, the study aims to incorporate multiple factors simultaneously and develop efficient evacuation routes. This approach enhances the effectiveness of evacuation planning by considering the spatial layout of the network and the constraints imposed by time and capacity limitations.

2.6 Subspace Occupancy Data

Once the subspace mentioned in *section 2.4* is established, the occupancy of each subspace is calculated. In the context of estimating people's flow movement during emergencies, it is assumed that spaces or rooms are fully occupied, reaching their maximum capacity. The estimation of occupancy for each space is based on the unit area required per person. This approach aligns with the method described in (Shreya and Rajan, 2023) for determining the subspace occupancy of each space. By considering these factors, the study aims to accurately estimate the outcome of the people's flow movement in emergency scenarios.

3. DATA

3.1 IFC, 3D to 2D building drawing.

A Building with a total *carpet area* of 102.349 m² is considered here. Furniture occupies an area of 19.569 m² and are placed across the different spaces. The building floor plan consists of six rooms with windows. Furniture such as bed, side table, cupboard, table, chairs, sink, bathroom installations, and sofa set, are used to indicate the obstacles in the path within the room space. The plan also includes the ceiling lamps that are not considered in this study. The building floor plan was exported into an IFC model with all the interior details. This was further used for semantic modeling. *Figure 1* shows the 3D building drawing with interior objects. The 2D building floor plan (*Figure 8*) clearly indicates blocked windows and area, nodes denoting subspaces, network connection from door to door, subspaces to door, and subspace to the window. The extra nodes present in the floor plan, as described earlier, are also illustrated here. Notation of the edges used is similar to that presented in (Shreya and Rajan, 2023).

3.2 Applications Used.

The application that is used for this work is shown in the flow diagram in *Figure 9*. The Revit 2022 with IFC 3D building floor plan was exported to Sketchup pro-2021 software where the tool was used to tag each and every component of the 3D building to make it compatible with CityGML. The SketchUp file with extension as. skp was added to FME workbench 2022.21 software as a reader file. The FME workbench was generated using the transformers as discussed in *section 2.1.3*. Transformers were used to give an output file as gml including building parts, building installation, building furniture, roof surface, and ground surface. The FZK viewer was used to visualize the GML model with different elements. Finally, the 3D building floor plan was exported as a flattened 2D floor plan, and calculations were made in QGIS software.

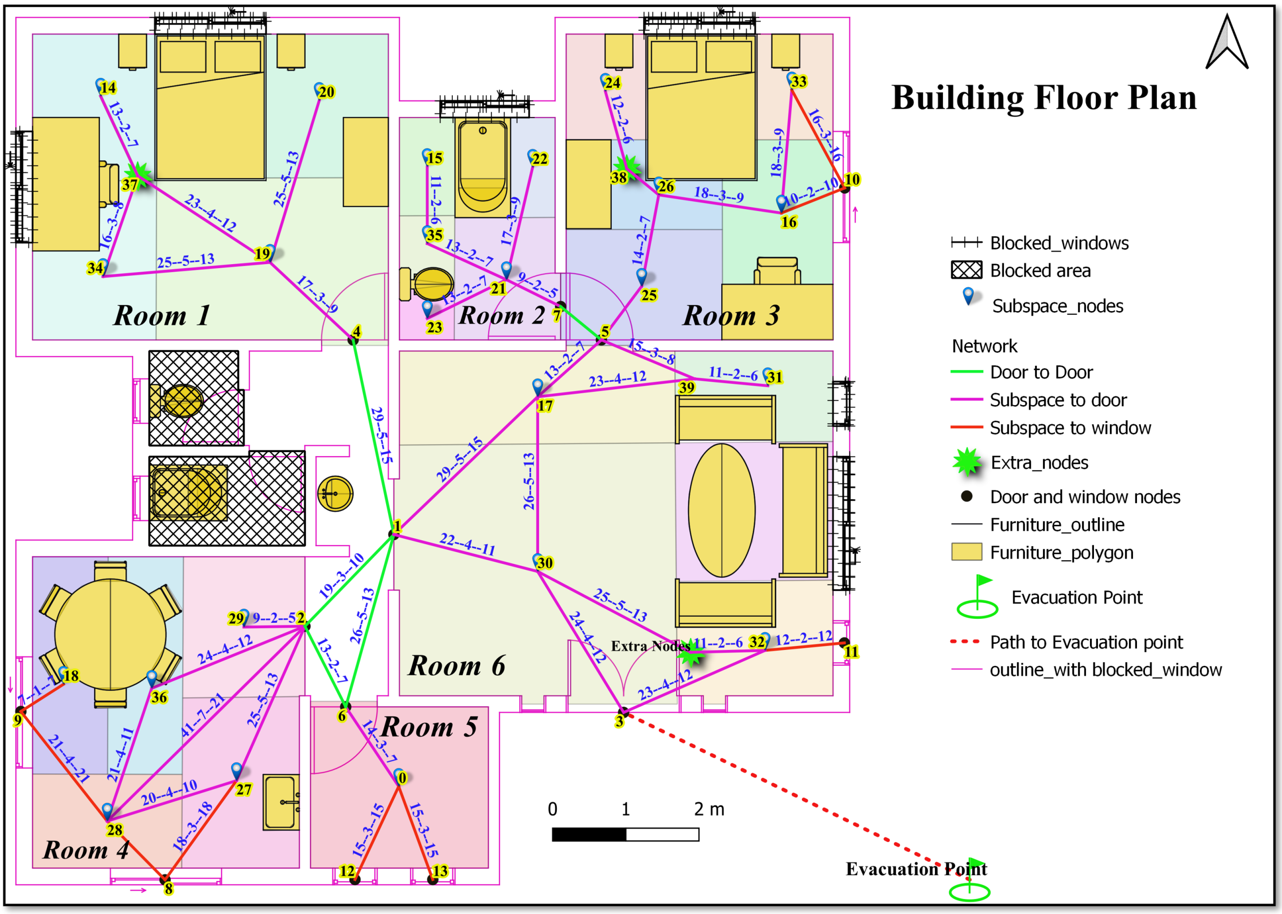

Figure 8. Building Floor plan with obstacles

3.3 Subspace occupancy

The generation of the subspace is described in *section 2.4* and the occupancy is calculated based on the area occupied by the occupants. The subspace with the occupants is mentioned in *Table 1*.

4. RESULT

4.1 Case 1: Base Scenario with obstacles considering one exit under static condition

The base scenario with obstacles shows the evacuation of occupants through the main exit. This is considered a static case, as there is no change in the network graph generated. Every subspace centroid has an unobstructed view of the door. There are 70 occupants because the area of space decreases as the obstacles such as furniture are placed in the space model. The area unoccupied by an obstacle gives the count of occupants as seventy so the calculation here is performed for seventy occupants only. Figure 10 shows the fluctuation in the graph as the door capacity is two and the delay occurs because of the position of an obstacle. The time taken for the seventieth person to exit is 215 seconds. While the last occupant takes only 194 seconds to exit in the base case without obstacles to exit.

4.2 Case 2: Multiple exit Scenario under the dynamic condition with obstacles.

During an evacuation, there can be multiple exits other than the main door. Evacuation through windows can also occur as there are some spaces that have two exits, such as room numbers 3, 4 5, and 6 as shown in *Figure 8*. The procedure for calculation and assumptions are based on (Shreya and Rajan, 2023). The assumption of blockage due to some material misplaced during the rush is made at node 5, *Figure 11*. In this dynamic condition, there is a possibility of any internal exit or node within the network becoming nonfunctional or acting as an obstacle, impeding free movement. If a blockage occurs at a particular node, it hinders the occupants' ability to evacuate through the main door. In this illustrated case, this needs the opening of an alternate exit - a window of Room 3. The process of finding a new evacuation path occurs dynamically and can apply to other building space models as well. As illustrated in *Figure 10*, due to delays, it takes approximately 996 seconds to evacuate the seventieth person. The abrupt increase at the green point **(Alternate exit 57, 308)** in *Figure 10* is attributed to the creation of a new opening within the room.

4.3 Case 3: Adverse case with critical node blockage under static and dynamic condition

An adverse case with critical node blockage indicates that the behavior of people changes when there are no other exits present for some of the occupants in the space model. In such cases, the people remove or drag the obstacles that can be removed or dragged easily. One such case has been shown here in this paper refer to *Figure 12*. The window node 12 here is replaced as an emergency door to see what happens in this case. Due to the placement of the emergency door at node 12, occupants from Room 4 pass through the emergency door at node

Figure 9. Flow diagram of the applications used.

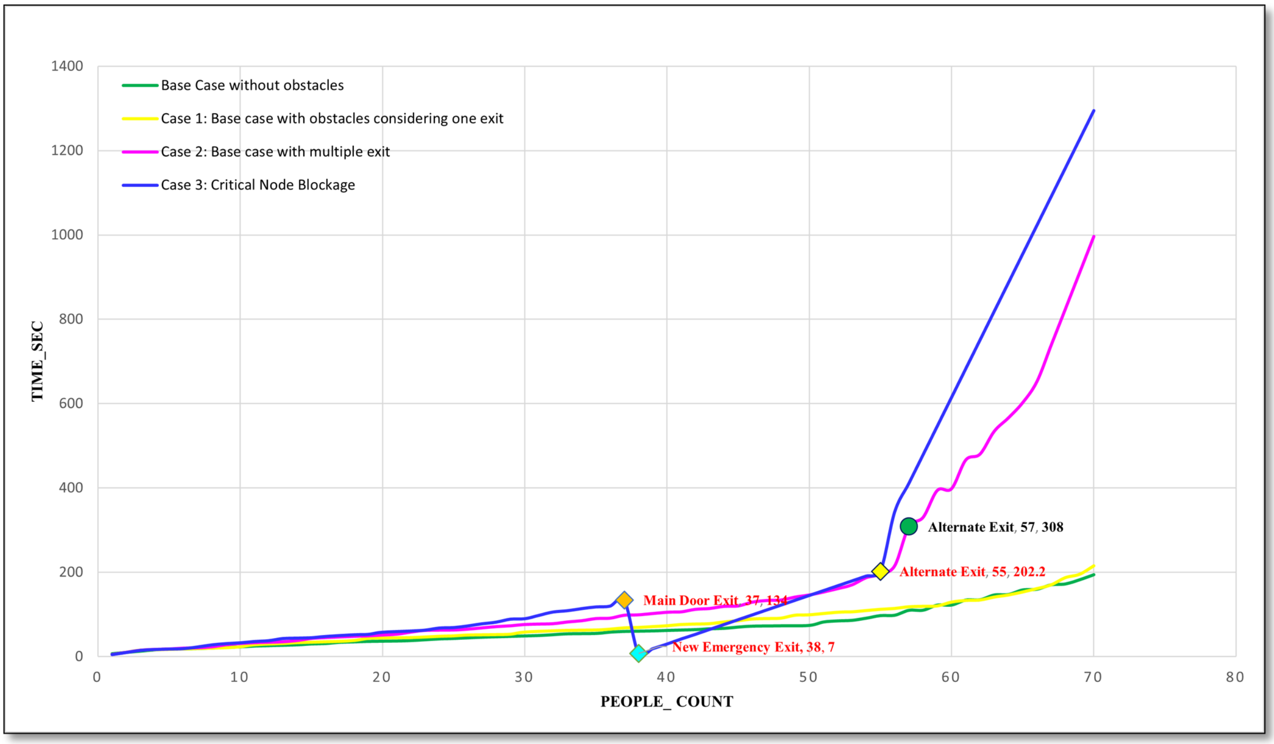

Figure 10. People Count v/s time graph for Base case, case 1, case 2, and case 3.

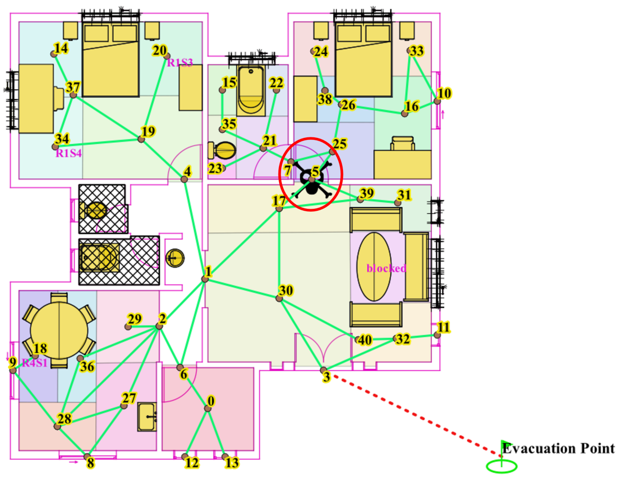

Figure 11. Floor plan with blocked node.

12. It takes around 1295 seconds for the last person to be evacuated. This delay has large values due to furniture removal/drag time and path length taken from one room to another. In this instance, the graph depicted in *Figure 10* displays a sudden decrease and subsequent increase in both the static and dynamic scenarios. The drop is observed at the orange point **(Main door exit 37, 134)** as a result of the introduction of the **(New Emergency exit 38, 7)** at the Cyan Point. However, the graph rises again at the Yellow point **(Alternate exit 55, 202.2)** due to the opening of windows and the removal of obstacles in Room 1.

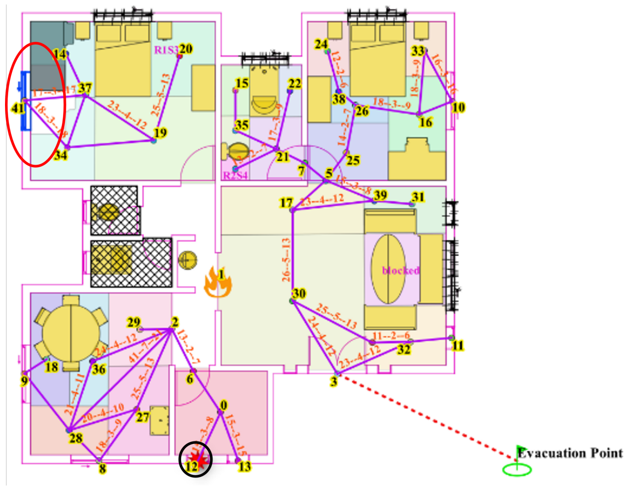

Figure 12. Floor plan with removable furniture and alternate exit as a door.

Number of occupants based on sub-space model			
Notations	Subspace	Occupancy Count	Total occupants
Room 1	R1S1	2	**15**
	R1S2	7	
	R1S3	3	
	R1S4	3	
Room 2	R2S1	1	**5**
	R2S2	2	
	R2S3	1	
	R2S4	0	
	R2S5	1	
Room 3	R3S1	3	**9**
	R3S2	1	
	R3S3	3	
	R3S4	1	
	R3S5	1	
Room 4	R4S1	2	**13**
	R4S2	3	
	R4S3	3	
	R4S4	3	
	R4S5	2	
Room 5	R5S1	2	**5**
	R5S2	2	
	R5S3	1	
Room 6	R6S1	5	**23**
	R6S2	13	
	R6S3	2	
	R6S4	3	

Table 1. Subspaces with their number of occupants.

5. DISCUSSION

The objective of this paper was to compare a base scenario without obstacles to a current approach with obstacles. The findings indicate that the presence of an obstacle resulted in only a marginally higher evacuation time of 215 seconds for 70 occupants, compared to 194 seconds in the base scenario with clear spaces and no obstacles. This slight increase is due to longer path lengths within the sub-spaces due to the presence of obstacles. In case 2 where the space has multiple exits, such as doors, windows, or emergency exit doors. where the obstruction led to disconnection, the evacuation time increased significantly to 996 seconds, almost four times the evacuation time in the base scenario. Additionally, in an adverse scenario involving critical node blockage, the time taken for the last person to evacuate was approximately 1295 seconds, representing a 6x increase compared to the base scenario with obstacles. Re-routing considerations were taken into account, as the extra time taken by individuals to find alternative exits influenced the evacuation process. While (Lui et al, 2011) utilized a door-to-door evacuation model that incorporated geometric and semantic information to provide evacuation instructions to individuals, the proposed work here takes the ”as-is” condition of the spaces and the obstacles to evaluate the evacuation outcomes. (Lui et al, 2016) used a static graph and focused on identifying the best-known path using an algorithm. It is important to acknowledge that all these approaches are graph-based. However, a key challenge with the existing approaches lies in the fact that the graph remains the same regardless of changes occurring within the environment such as delays in between the path.

Furthermore, the comparison in this study does not directly focus on routing computation, as the number of routes considered is not very high. Instead, we focused on the time it would take for individuals to exit, which is dependent on people's flow and rate. This approach accounts for a lag between individuals during the evacuation process. By utilizing a graph-based model, changes in occupancy could be accounted for, making our approach more robust.

6. CONCLUSION

The work proposed a 3D to 2D structure conversion for evacuation modeling of the occupants based on subspaces that emerge due to obstacles present in the room spaces. These subspaces provide a clear line of sight to the exit and hence are part of the evacuation path. Compared to the base scenario without any obstacles, the current approach with obstacles, where the door was the primary exit, took only a marginally higher time of 215 seconds for 70 occupants, as compared to 194 seconds with clear free spaces, due to the increased path lengths over the subspaces. There are some possible cases where the space has two exits such as a door, window, or emergency exit door. When there is an obstruction leading to a disconnection of a part of the network, an alternate exit like the window is used for some spaces. The results show that in case 2, there is almost a 4x rise in evacuation time. This indicates that alternate exits have to be properly planned to reduce the delays in accessing these alternate exits and the people flow rate across them. In the adverse case with critical node blockage, the model shows that the evacuation time can be as high as 6x for the last person to exit, clearly indicating the challenges that dynamic changes can create due to the emergence of new obstacles or blockages and the need to re-route the evacuees. Further work is needed to integrate the proposed method with seamless capture and integration of the static and dynamic components of the indoor spaces and develop a proper occupant-friendly information dissemination model.

7. REFERENCE

Shreya, . and Rajan, K. (2023). Modeling & Simulating the Evacuation of a Building Based on Building Floor Plan and Evacuation Strategies. In Proceedings of the 9th International Conference on Geographical Information Systems Theory, Applications and Management, ISBN 978-989-758-649-1, ISSN 2184-500X, pages 230-236. DOI: 10.5220/0011992600003473

Khan, A. A., Donaubauer, A., Kolbe, T. H. (2014, November). A multi-step transformation process for automatically generating indoor routing graphs from existing semantic 3D building models. In Proceedings of the 9th 3D GeoInfo Conference.

Liu, L., & Zlatanova, S. (2011). A” door-to-door” path-finding approach for indoor navigation. Proceedings Gi4DM 2011: GeoInformation for Disaster Management, Antalya, Turkey, 3-8 May 2011.

El-Mekawy, M., b Anders, Ö., & c Ihab, H. (2012). An evaluation of IFC-CityGML unidirectional conversion. International Journal of Advanced Computer Science and Applications, 3(5).

Zhu, J., Wright, G., Wang, J., & Wang, X. (2018). A critical review of the integration of geographic information system and building information modelling at the data level. ISPRS International Journal of Geo-Information, 7(2), 66.

Ma, J., Jia, W., & Zhang, J. (2017, May). Research of building evacuation path to guide based on BIM. In 2017 29th Chinese Control And Decision Conference (CCDC) (pp. 1814-1818). IEEE

Liu, C., Mao, Z. L., & Fu, Z. M. (2016). Emergency evacuation model and algorithm in the building with several exits. Procedia Engineering, 135, 12-18.