A Real-time Path-planning Model for Building Evacuations

Farid Mirahadi[a] and Brenda McCabe[a]

[a]Department of Civil Engineering, University of Toronto, Canada
E-mail: f.mirahadi@mail.utoronto.ca, brenda.mccabe@utoronto.ca

Abstract –

Simultaneous evacuation is the most widely used evacuation strategy in buildings. However, there are other evacuation strategies that might lead to safer outcomes if selected appropriately. Different forms of evacuation result from applying time delays to phased evacuation or altering path planning. The best strategy for evacuation depends on the characteristics of the building and the circumstances of the particular emergency. A real-time evacuation path-planning model that identifies the fire hazard and proposes the best strategy of evacuation during the emergency can reduce risk and improve safety. In this paper, a model is proposed to find the safest strategy of evacuation based on the current state of the building and the emergency case. The model focuses on fire emergencies, as they are the dominant cause of fatalities in buildings compared to other types of natural and manmade disasters. The proposed model first defines a risk factor for each compartment based on the location of fire and then calculates the lowest risk path using Dijkstra algorithm. The path-planning runs on the geometric network graph (GNG), which is generated from the IFC file of the building. Furthermore, unexpected events during evacuation, e.g. another source of fire, can force the system to search for another strategy. Herein, a model is designed to monitor the building in real-time and in case of any unexpected event, changes the evacuation plan accordingly. The case study shows that the proposed model for real-time evacuation management can significantly enhance the safety level of evacuation compared to the conventional simultaneous evacuation process.

Keywords –

Evacuation; Dijkstra; Route risk index; Geometric network graph; BIM; IFC; Fire safety

1 Introduction

Decision-making and coordination for evacuation processes are normally managed by a human emergency commander. This person has to comprehend the status of the building, estimate the distribution of the residents, recognize the type and the location of the hazard, decide the best strategy of evacuation and transfer the message to occupants. The procedure gets even more complicated when the commander has to monitor the building and the evacuees during evacuation and improvise solutions for unexpected situations that differ from the forecasted plan. Such complexities in decision making, data interpretation, time constraints, and inevitable human errors highlight the benefits of using automated systems to inform the decision making processes.

Computer simulation can stochastically imitate the course of events during emergencies. Today, crowd simulation approaches make it possible to model the movements and the behaviors of residents during an evacuation. Moreover, the growth pattern of the hazard itself, in cases such as fire, can be modeled and forecasted via software applications and simulators. Such tools enable us to examine a specific building design and evaluate the safety of evacuation processes in that facility. By developing different emergency scenarios and simulating them, a computer-based decision support system can determine the best strategy of evacuation.

However, the main goal of evacuation management process is to facilitate decision making during the emergency event. It is no surprise that the real evacuation process will be different from the simulated event in some aspects. People might escape by different routes, the number and distribution of evacuees could vary, congestion could occur, or a new threat could be added to the original one. These deviations from the simulated scenario need an efficient model that performs path-planning within the tight time limits. During evacuation, there is not sufficient time for running multiple iterations of a fire simulation or a detailed crowd simulation. This highlights the need for a real-time path-planning model that responds to changes in emergency situations and promptly generates new plans that secure the safety of the occupants. For a long time, evaluation of evacuation processes has been measured based on finding plans that result in the shortest exit routes or shortest time of evacuation. However, a successful path-planning model needs to assess and rank the strategies based on the level of safety and the risks to evacuees' lives, and not only the time of evacuation.

The objective of this paper is to present a real-time path planning model for building evacuations during fire emergency events.

2 Literature Review

The main goal of an evacuation management system is real-time path-planning of escape routes in a building. Path-planning requires a measure based on which the privilege of each specific route can be quantified. In fire emergencies, a combined temperature-smoke risk index measured by heat and smoke detectors has been used to find the optimal evacuation routes. Finally, the optimal routes were transmitted to occupants through smartphone-based devices [1]. The study did not provide enough information about the computational load, the time required for modeling, and how they addressed the challenges of using positioning techniques in indoor spaces. In another research, predictions of stochastic evacuation models were used to generate the optimal evacuation routes in real-time [2]. The framework lacked explicit attention to hazard detection and threat propagation. Dynamic exit signs were proposed for use in a smart emergency management system for tall buildings to guide evacuees towards the safest routes within a proposed geometric architecture [3]. Researchers also used information regarding ignition points, locations of trapped occupants, and locations of firefighters to help firefighters find the optimal route to access trapped person [4].

Real-time path-planning requires accurate and updated information about the physical and functional characteristics of the building. This information is collected and organized in building information model (BIM). BIM can be used to construct a door-to-room connectivity graph that facilitates path-planning in a building [5]. The interoperability of industry foundation classes (IFC) motivated researchers to extract the topological connectivity graph of the building directly from IFC data structure. This approach also uses doors to recognize the connected spaces of the building [6]. The transformation from BIM to a network graph allows the use of graph exploration algorithms for path-planning. Medial Axis Transform algorithm can also be used to produce the geometric topology network (GTN) of a building [7]. Medial axis of a polygon is the collection of all points inside the polygon having more than one closest point on the polygon's boundary.

Traditionally, the main purpose of egress design and path-planning has been to find the shortest path of evacuation. The original problem of finding the shortest path between two points has been solved based on graph exploration methods. Dijkstra algorithm is a graph search algorithm that finds the global shortest path between two nodes of a graph [8]. The cost function of the algorithm in its simplest form is based on the distance between the nodes. A* is another widely used algorithm, but instead of performing a global search, it contains a heuristic function. Similar to Dijkstra, it calculates the cost of traversed paths, while the heuristic leads the search directly towards the goal [9]. Compared to Dijkstra, the computational load of A* is much less and consequently the algorithm is faster. However, the solution is not necessarily globally optimal. In emergency management, Dijkstra has been used to find the shortest rescue path to victims [10]. A combination of BIM and Dijkstra was used for finding optimal evacuation/rescue routes [11].

The quality of evacuation processes has been primarily measured by the time of evacuation. The total evacuation time, average evacuation time of evacuees, and total evacuation time of each evacuee are some examples of this measure. However, real-time management of crowds during emergencies demands that people be guided away from the hazard along the safest possible route. In other words, finding the shortest path cannot necessarily guarantee a safe evacuation.

Route Risk Index (RRI), a risk index designed to quantify the risk of each egress route, was proposed to allow comparison between egress alternatives [12]. This index considers the length of evacuation and proximity to hazards in its calculation. Equation (1) shows the formula of RRI.

$$RRI = \int_{t=0}^{T} \frac{1}{ALET_{p(t)}} \; dt \qquad \text{Equation (1)}$$

where:

$ALET_{p(t)}$ (Available Local Egress Time) is the time it takes for a fire to reach $p_{(t)}$;

$p_{(t)}$ is the location (x,y,z) of an evacuee at time t;

T is total travel time of an evacuee from their starting location to the end of the egress route.

Given Equation (1), the risk of being at point $p_{(t)}$ is equal to 1/ALET, which is integrated over time to determine the risk for the entire evacuation path.

3 Model Development

The flowchart of the proposed model is presented in Figure 1. It is worth noting that since fire emergencies are the most frequent causes of evacuations in buildings, this model focuses on fire as the hazard. In this model, BIM is used to acquire information related to functional and physical characteristics of the facility. The BIM model is transformed to IFC data structure and geometric network graph (GNG) of the building is generated from it. It is assumed that an existing decision support system, such as EvacuSafe [12], already selected the best strategy of evacuation based on the current state of the building and the occupants at the beginning of the emergency. Such decision support system has to conclude based on precise fire simulation and crowd simulation. Here, the main goal

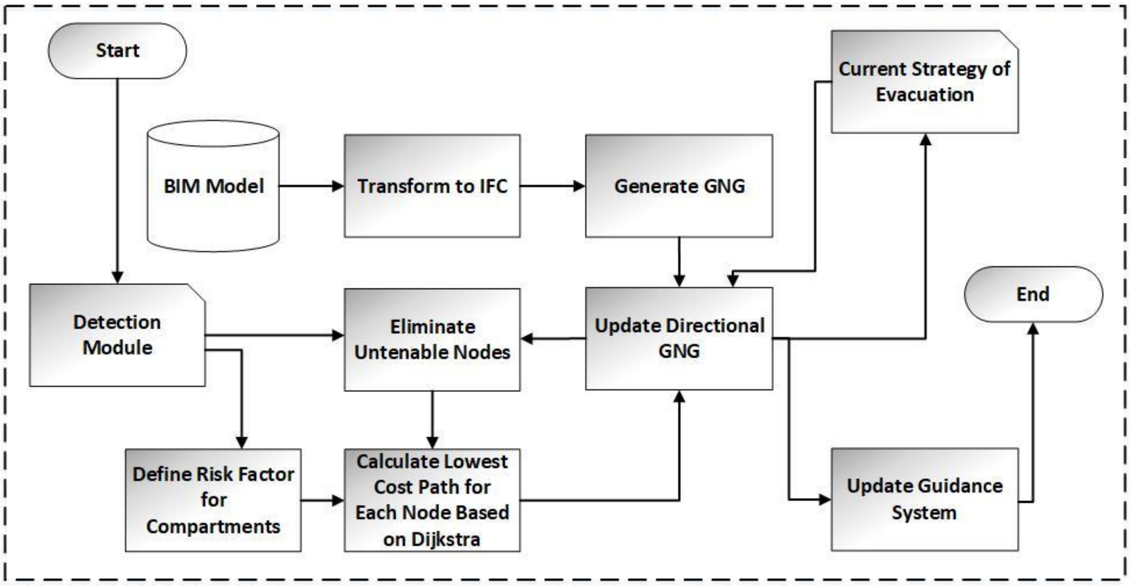

Figure 1. Flowchart of the proposed model

is to update the guidance system during the course of the emergency as the real sequence of events unfold. Fusion of the selected strategy of evacuation with the GNG of the building produces a directional network graph. Once the detection module locates the exact coordinates of the hazard, the associated node in the GNG is eliminated. A risk factor is assigned to all of the compartments/nodes in the building and Dijkstra algorithm is applied to find the updated safest path of evacuation for each node. The guidance system can then be updated.

3.1 GNG

A 3D network graph is required to represent the navigable connections between the spaces in the building. Each node stands for a space and each edge represents a connection between the spaces. This graph also has to contain information about the distances and directions of the pathways.

The graph is defined in continuous 3D space. Each node is positioned at the center of each space (room), and the length of each edge is proportional to the distance between the nodes. In order to generate this graph automatically, two sets of information are required: 1) The data related to the available connections between the spaces; and, 2) The data related to the location of each node in the space and the distance between the nodes. IFC data structure can provide this information and remain independent of a specific BIM file extension, but requires some explanation.

ifcRelSpaceBoundary was first introduced in IFC release 1.5 and was later modified in the 2X release. This entity determines the physical or virtual relationship of each space with its surrounding elements. Two attributes of this entity are *RelatingSpace* and *RelatedBuildingElement*. *RelatedBuildingElement* specifies the elements, including walls, doors or virtual elements that are immediately connected to that space. *ifcDoor* and *ifcVirtualElement* matter here. In this terminology, a virtual element is a delimiter between the rooms or spaces. This element does not exist in the built environment and its only purpose is to allow decomposition of one space into smaller segments. The IFC file is parsed and a query is run to filter the spaces sharing common virtual elements or doors. Therefore, spaces sharing the same element of these two types are connected to each other. Gradually the graph of the entire building is constructed. This graph contains information about the characteristics of all connections between the spaces.

The other element that needs to be extracted is the center of each space. *ifcSpace* has two subtypes of *IfcProductDefinitionShape* and *ifcLocalPlacement*. *ifcLocalPlacement* defines the placement of the element, i.e. space here, in the 3D space. *IfcProductDefinitionShape* describes the physical or topological representation of the product. What matters here is the footprint of the space mirrored on the navigable surface. Each space's boundaries are represented by polylines and composite curves [13]. The type of this mirrored shape on the surface combined with the information related to the coordinates of the boundaries facilitate calculation of the centroid of each space. The combination of the network graph generated from navigable connections between the spaces and the centroids of the spaces generates a GNG in which the length of edges and coordinates of nodes exactly imitates the building dimensions in reality.

3.2 Eliminate Untenable Nodes and Request Updated Path-planning

The GNG prepared in the previous step shows the navigable skeleton of the building. However, it does not present any information regarding the desired movement of the evacuees in the building. An evacuation strategy selection system, such as EvacuSafe, has the ability to select the safest strategy of evacuation based on the first signal from fire alarm system [12]. It positions the location of the hazard and then based on integrated fire and agent based crowd simulations prescribes the safest strategy for evacuation. Merging the best strategy with the GNG generates a directional GNG, which specifies the safest direction of evacuees' movements by inserting arrows on the edges of the GNG.

GNG is only the visual representation of space connections in the building. An adjacency matrix is the mathematical representation of the GNG. An adjacency matrix is an *n-by-n* matrix in which n is equal to the total number of nodes and is filled with 0 and 1 values. A value of 1 indicates that the pair of nodes are connected to each other. In the case of directional graphs, only one element of the pair is marked as 1.

Once a fire case is detected in the building, the fire alarm module sends a signal containing the location of the fire. The node in which the fire is located is eliminated from the directional GNG. As the fire expands, more nodes are eliminated. These changes in the GNG forces recalculation of the safest path for each node. Since this happens during an emergency, there is no time to redo the fire simulation or crowd simulation. As such, another algorithm is required to quickly calculate the risk of available exit paths so that the crowd can be redirected through the safest ones.

3.3 Risk Factor of Compartment

As the fire is initiated in the building, each spot in the building is subject to a level of risk. Herein, the risk factor is defined based on philosophy of RRI. As mentioned previously, 1/ALET specifies the risk of being present at each spot in the building. This risk factor is distributed based on the time that each space can stay tenable from the evidence of fire. The risk factor provides a basis to know which zones are in higher danger compared to the others. However, ALET has to be computed via fire simulation. Fire simulation is computationally heavy and cannot be performed within the tight time limits of an emergency situation. Therefore, before the emergency occurs, the initiating point of fire has to be placed at all of the compartments of the building in fire simulation, one after the other. The results of simulation reports the ALETs for all of the compartments associated with each initiating point of fire. In this way, each node of the building is assigned by a risk factor. If another fire occurs concurrently in the building, the risk factors add up in each node.

3.4 Dijkstra Algorithm

The Dijkstra algorithm is a widely used graph search algorithm in navigation and robotics. The algorithm can be modified to calculate the cost of each path instead of the number of steps taken. In this research, two factors influence the safety of evacuation. The first is the travel time of the evacuee. The second is the path's proximity to the fire. Proximity is represented by the risk factor RRI. Based on these requirements, the cost function for the traversed path between two nodes is defined in Equation 2 using the Euclidean distance algorithm.

$$f(x) = R \times \frac{\sqrt{(x_2-x_1)^2+(y_2-y_1)^2+(z_2-z_1)^2}}{\bar{V}} \qquad (2)$$

where:

x_1, y_1, z_1 are the coordinates of the start node

x_2, y_2, z_2 are the coordinates of the end node

R is the risk factor of the edge connecting node 1 to node 2

$\bar{V}$ is the average speed of evacuees

Since the risk factors were previously defined for all of the nodes, the risk factor of the edges are calculated based on the average of two connecting nodes.

In Dijkstra, the start node is tagged first. Then, the cost of travel to all the connected nodes is calculated. The node with the smallest value is tagged. Then, the cost of travel to all of its connected nodes are calculated. The cost of unvisited nodes is updated if it is less than their previous cost. This process continues until the end node is met.

Large facilities and high-rise buildings typically have multiple exit doors. Therefore, there are multiple goals in this pathfinding problem. The Dijkstra algorithm is therefore solved for multiple exit goals and the minimum of those is the final solution.

4 Case Study

A hypothetical floor plan is used for implementation of the proposed model. The upper part of Figure 2 shows an isometric view of the three floor building with three stair cases that connect each floor level. The lower part of Figure 2 shows the floor plan, which is standard for each floor. The 3 exit doors are at the bottom of each staircase.

The BIM model of the building is translated to IFC data structure. IFC file is parsed and GNG is extracted. It is assumed that an analysis has been performed and it has been concluded that the best strategy of evacuation is simultaneous evacuation. Simultaneous evacuation is the most prevalent strategy in which all of the occupants are asked to exit at once.

Based on simultaneous evacuation, the directional graph of the building is generated and shown in Figure 3. Note that the exit doors on the ground level are shown as red circles. The length of each edge is shown in meters, and the risk factor (R) at each node is shown in blue.

In this case, occupants in each compartment on the 2nd and 3rd floors can take all three exits and the staircases are one-way downward. That is why the edges on the 2nd and 3rd floors do not have directional arrow and the edges in the staircases do.

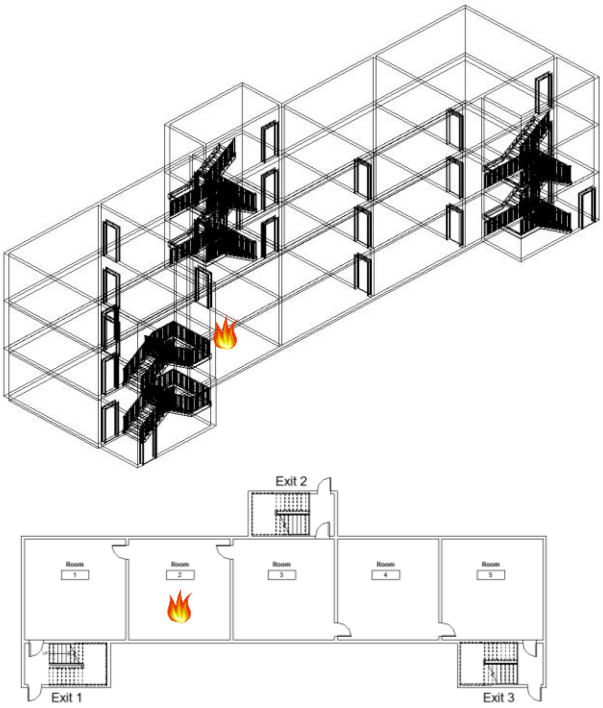

Figure 2. Isometric view and floor plan of the case study

For this research, the widely known Fire Dynamics Simulator (FDS) [14] developed by National Institute of Standards and Technology (NIST) is used to simulate the fire initiation and expansion. It is assumed that the fire-initiating element is a piece of household furniture, represented by a 1 cubic meter block of wood. The early alert of un-tenability or an approaching fire has a threshold defined by the presence of smoke that results in a visibility obstruction of 11%. Due to the computational heaviness of FDS, the fire simulations are run for a large number of scenarios during the evaluation planning stage, i.e. well before the emergency happens. Initiating point of fire is placed in all of the nodes and un-tenability times of the rest of the nodes are calculated. Simulation results are stored for access during an emergency situation.

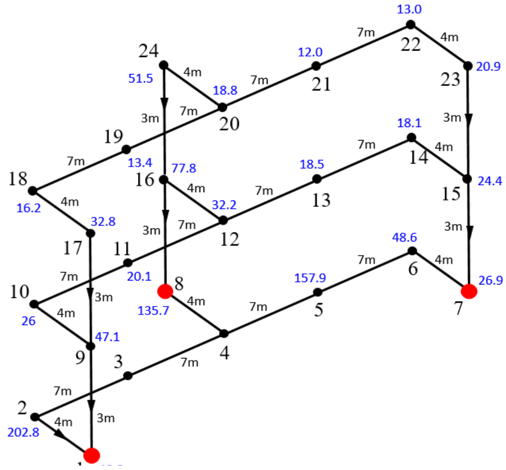

Figure 3. Directional GNG of the building with risk factors

Let's assume that the fire is initiated in node 3 on the first floor and after a specific amount of time, the fire expands. Smoke is quickly detected in node 4. It is prudent at this time to test if simultaneous evacuation is still the best strategy for evacuation.

The un-tenability condition defines ALET in Equation 1. Using this equation, the risk factor of each node is the summation of 1/ALET of the fire initiated in node 3 and node 4. The risk factor of each edge is then defined based on the average risk factor (R) of the nodes at two ends of the edge. The average speed of evacuees is assumed 1.35 (m/s) based on a study on commuters with varying genders and ages [15]. Dijkstra is later run for each node based on the cost function of Equation (2). Table 1 tabulates the result of this calculation. The 'selected route' comprises the path, as labeled in the directional GNG. The lowest cost path for each node is listed. The cost of the *shortest* path is also reported for the purpose of comparison. In 18 of the 24 nodes, the shortest path is the lowest cost path. In six cases, however, the shortest path is not the safest path. These are shown shaded in red.

For example, the lowest cost path from node 12 to an exit goes through nodes 12-13-14-15-7. Its cost is $346.5*10^{-4}$. Note that this path takes occupants to the exit at node 7 and not to the closer node 8 exit, which has a higher cost ($400.2*10^{-4}$) because it is near the fire at nodes 3 and 4. Merging the results in Table 1 with the GNG will result in the directional GNG shown in Figure 4. Note that all of the edges in the graph are now directional.

Table 1. Result of Dijkstra graph search

Node	Selected Route As per Eq.2 (Lowest Cost Path)	Cost of Selected Route ($\times 10^{-4}$)	Cost of Shortest Path ($\times 10^{-4}$)
1	1	0.0	0.0
2	2 1	372.9	372.9
3	3 2 1	Untenable	Untenable
4	4 8	Untenable	Untenable
5	5 6 7	647.5	647.5
6	6 7	112.0	112.0
7	7	0.0	0.0
8	8	0.0	0.0
9	19 1	106.6	106.6
10	10 9 1	214.9	214.9
11	11 10 9 1	334.4	870.2
12	12 13 14 15 7	346.5	400.2
13	13 14 15 7	215.0	746.7
14	14 15 7	120.0	120.0
15	15 7	57.0	57.0
16	16 8	237.3	237.3
17	17 9 1	195.3	195.3
18	18 17 9 1	268.0	268.0
19	19 18 17 9 1	344.7	913.2
20	20 21 22 23 15 7	302.1	485.1
21	21 22 23 15 7	222.2	787.2
22	22 23 15 7	157.6	157.6
23	23 15 7	107.4	107.4
24	24 16 8	380.9	380.9

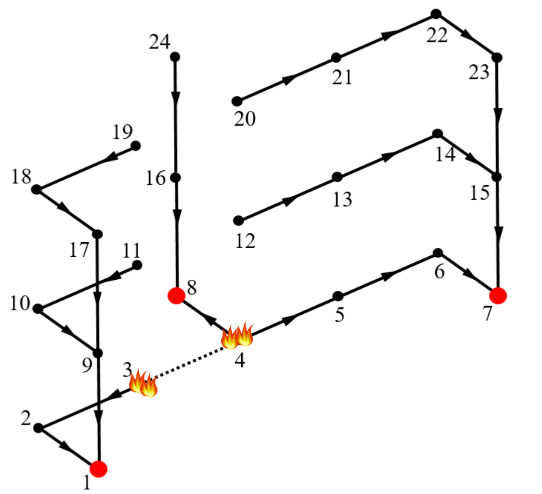

Figure 4. Resulted directional GNG from real-time path-planning model

Figure 4 graphically shows the safest evacuation routes for the current situation. Compared with simultaneous evacuation where evacuees are free to select any route, this graph recommends the evacuees in each node move away from the middle exit. It is interesting how adjacent compartments of 19-20 and 11-12 are directed to take separate egress routes. This emphasizes the point that the shortest exit path is not necessarily the safest or the most efficient option of evacuation.

The graph exploration and cost optimization of the proposed method, which must be performed during evacuation process, took less than 1.0 second on a Core i-7 dual core 2.7 GHz. This quick computation time satisfied the need for an algorithm that could reliably perform during the evacuation process. It was supported by the thorough exploration of emergency scenarios that performed during evacuation planning when the building was first occupied.

5 Conclusion

In this paper a real-time path-planning model for building evacuation was proposed. The model used IFC to automatically generate a GNG of the building. The Dijkstra method was modified and applied for exploration of the lowest cost path. The cost function of Dijkstra was defined based on combination of the proximity to the source of the fire and the length of the escape route. The risk factor representing the proximity to the source of fire was adopted from the RRI concept.

Finally, a hypothetical case study showed that the proposed model can successfully operate within tight time limits of an evacuation process. The result also showed that the conventional notion of "the shortest path is the safest path" may not always be true.

Future work in this research requires a modification to the cost function in order to include other influential factors in path-planning. Incorporating directional bias of the movements, congestion, speed variety in different zones, blockage caused by structural collapse and complexity of the routes can significantly enhance the accuracy and confidence of path-planning, which leads to a safer evacuation process.

References

[1] Lujak, M. Billhardt, H. Dunkel, J. Fernández, A. Hermoso, R. and Ossowski, S. A distributed architecture for real-time evacuation guidance in large smart buildings. *Computer Science and Information Systems*, 14(1), 257-282, 2017.

[2] Cuesta, A. Abreu, O. Balboa, A. and Alvear, D. Real-time evacuation route selection methodology for complex buildings. *Fire Safety Journal*, *91*, 947-954, 2017.

[3] Mirahadi, F., McCabe, B., Shahi, A. Smart Disaster Management System for Tall Buildings. In

Proceedings of CSCE 2017, Canadian Society of Civil Engineers, Vancouver, BC, Canada, 2017.

[4] Chou, J. S. Cheng, M. Y. Hsieh, Y. M. Yang, I. T. and Hsu, H. T. Optimal path planning in real time for dynamic building fire rescue operations using wireless sensors and visual guidance. *Automation in Construction*, *99*, 1-17, 2019.

[5] Yan W., Culp C. and Graf, R. Integrating BIM and gaming for real-time interactive architectural visualization. *Automation in Construction*, 20(4): 446-458, 2011.

[6] Eastman, C. Lee, J. M. Jeong, Y. S. and Lee, J. K. Automatic rule-based checking of building designs. *Automation in construction*, 18(8): 1011-1033, 2009.

[7] Taneja, S. Akinci, B. Garrett, J. H. Soibelman, L. and East, B. Transforming IFC-based building layout information into a geometric topology network for indoor navigation assistance. In *Computing in Civil Engineering*, 315-322, 2011.

[8] Dijkstra, E. W. A note on two problems in connexion with graphs. *Numerische mathematik*, 1(1): 269-271, 1959.

[9] Hart, P. E. Nilsson, N. J. and Raphael, B. A formal basis for the heuristic determination of minimum cost paths. *IEEE transactions on Systems Science and Cybernetics*, 4(2): 100-107, 1968.

[10] Wu, C. H. and Chen, L. C. 3D spatial information for fire-fighting search and rescue route analysis within buildings. *Fire Safety Journal*, 48: 21-29, 2012.

[11] Cheng, M. Y. Chiu, K. C. Hsieh, Y. M. Yang, I. T. Chou, J. S. and Wu, Y. W. BIM integrated smart monitoring technique for building fire prevention and disaster relief. *Automation in Construction*, 84: 14-30, 2017.

[12] Mirahadi F., McCabe B., Shahi A. IFC-centric performance-based evaluation of building evacuations using fire dynamics simulation and agent-based modeling. *Automation in Construction*, 101: 1-16, 2019.

[13] Lin Y. H., Liu Y. S., Gao G., Han X. G., Lai C. Y. and Gu M. The IFC-based path planning for 3D indoor spaces. *Advanced Engineering Informatics*, 27(2): 189-205, 2013.

[14] McGrattan, K. Hostikka, S. Floyd, J. Baum, H. R. Rehm, R. G. Mell, W. and McDermott, R. *NIST special publication, 1018-5: Fire dynamics simulator (version 5), technical reference guide.* NIST, 2007.

[15] Fruin, J. J. *Pedestrian planning and design.* Metropolitan Association of Urban Designers and Environmental Planners, 1971.