Tianjin Daxue Xuebao (Ziran Kexue yu Gongcheng Jishu Ban)/
Journal of Tianjin University Science and Technology
ISSN (Online):0493-2137
E-Publication: Online Open Access
Vol: 57 Issue: 02:2024
DOI: 10.5281/zenodo.10686214

A COMPREHENSIVE FRAMEWORK FOR FIRE EVACUATION MODELLING

MURNI FATEHAH ALIAS
PhD Student, Faculty of Computing, University Malaysia Pahang Al-Sultan Abdullah, Pahang, Malaysia.
Email: murnifatehah@gmail.com

AL-FAHIM MUBARAK-ALI*
Faculty of Computing, University Malaysia Pahang Al-Sultan Abdullah, Pahang, Malaysia.
*Corresponding Author Email: fahim@umpsa.edu.my

ROZLINA MOHAMED
Faculty of Computing, University Malaysia Pahang Al-Sultan Abdullah, Pahang, Malaysia.
Email: rozlina@umpsa.edu.my

Abstract

Strategic planning for the evacuation of occupants from buildings becomes crucial during disaster response operations, especially in the context of fire emergencies that pose a direct threat to human lives. This study addresses the specific challenges associated with fire evacuation in smart buildings within the Internet of Things environment. Despite their enhanced connectivity and accessibility, these buildings are still vulnerable to crises; therefore, efficient and swift occupant evacuation planning is necessary. Developing a successful evacuation plan for these situations calls for in-depth knowledge of smart building characteristics, evacuation factors, and skilful modelling. In order to overcome this problem, the research proposed a metamodel approach that serves as a modelling grammar and syntax for systematic design. The metamodel is constructed based on common evacuation model terminology and fire emergency variables. Through the use of a graphical editor and model transformation, the metamodel undergoes validation using a model-checking technique. In abstract modelling, this validation technique provides crucial insights into the accuracy and completeness of the metamodel in enhancing the resilience of smart building evacuation systems.

Index Terms: Disaster Response, Fire Evacuation, Formal Method, Metamodel, Metamodeling, Model-Checking, Smart Building.

1. INTRODUCTION

Buildings are designed as sustainable infrastructure for long-term functionality [1] and are inevitably vulnerable to the possibility of a fire incident. This incident could directly or indirectly impact its operation and pose serious risks to structure and human life [2]. In 2021 alone, Malaysia reported 35,902 cases of fire emergencies, with more than 75 people reported dead [3]. This alarming statistic indicates the urgent need for effective strategies for people’s safety. Moreover, the selection of building materials is crucial in determining the degree of destruction, with natural wood proven to suffer huge losses compared to densified wood or brick building material [4]. For this reason, better building design and materials are needed to help minimise the number of fatalities and level of property loss. The necessity for improved building design and material becomes more evident when considering the crucial connection between structural robustness and the evacuation process.

Evacuation is a process of removing any life, such as people, animals, or valuable property, from a place of danger to a safer place [5]. Some researchers also suggest that the evacuation concept is derived from another Latin word, evacuation, that gives the meaning of emptying, disappearing or evacuate, meaning emptying, disappearing [6]. Most fire evacuation plans are dependent on assumptions about occupant behaviour and static building layout [7]. Recently, advancements in smart building technology have changed how evacuation modelling is designed through their ability to predict occupant behaviour based on factors such as building layout, fire dynamics, occupant decision-making, and crowd behaviour [8].

The incorporation of smart building tech ologies into evacuation models, such as sensors, actuators, Internet of Things (IoT) devices and real-time analytics, presents an opportunity to improve and streamline the evacuation process [9]. However, existing models fall short in addressing the complexity of smart building systems, particularly the ability to exploit real-time data effectively, adapt to evolving evacuation conditions and comprehend the dynamic interactions between fire emergencies, occupants, and the smart building infrastructure. Furthermore, these existing models encounter challenges related to their flexibility, scalability, and accuracy.

To address these challenges, researchers in the field of building fire evacuation are turning to the concepts of metamodels. Metamodel serves as a higher-level model that defines modelling language by incorporating and encapsulating the essential aspect of another lower-level model [10]. Through this approach, metamodels offer a comprehensive and flexible framework for analysing complex systems, providing a solution to the limitations associated with existing evacuation models. Taking advantage of metamodel concepts, this research aims to develop a metamodel specifically designed for safety evacuation within smart building systems. This metamodel will bridge the gap between conventional fire evacuation models and the innovative smart building system paradigm, enabling the metamodel to integrate real-time data effectively and leverage the features of smart building systems, thereby enhancing the effectiveness of the evacuation process.

This research employed a fire emergency scenario within a five-floor residential smart building as a case study to assess the proposed metamodel. This emergency scenario incorporated a range of sensors, actuators and other smart building technologies. This case study served a dual purpose: firstly, to design a state-machine model for defining the proposed metamodel and secondly, to simulate the UPPAAL model for model checking evaluation. In summary, the main contribution of this paper lies in the development of the proposed safety evacuation metamodel that integrates a smart building system. Additionally, this research also offers perspective work on the validity of model-checking techniques for metamodel evaluation.

The rest of the paper is organised as follows. Section 2 introduces the related work. Section 3 describes the metamodel development method and Section 4 details the proposed metamodel. Section 5 explains the model checking evaluation, and finally, the conclusion is presented in Section 6.

2. RELATED WORK

Various strategies have been developed to improve the development of efficient evacuation plans in response to the growing interest in evacuation modelling. One such strategy involves the introduction of modelling language, SensorML, which provides a comprehensive framework for specifying processes related to sensors [11]. This language proves valuable in managing sensor information throughout the evacuation process. Another strategy was proposed by Kaito et al. [12], who introduced an emergency evacuation system that utilised wireless sensor network technology to gather real-time data on light statuses. Then, this information is subsequently transmitted to a control centre, which simplifies evacuation instructions, cuts costs, and improves overall stability. The study emphasises the idea that incorporating technology into the evacuation process not only enhances efficiency but also proves to be cost-effective.

Another impressive work is a comprehensive review done by Ronchi et al. [13] that explored the use of evacuation models to analyse relocation strategies and safety concerns in high-rise buildings. Similarly, Han et al. [14] also focused on high-rise structures, introducing real-time evacuation route planning in the event of a fire emergency. The study evaluates fire status to formulate effective evacuation strategies, emphasising the crucial role of planning staircases in high-rise buildings. Soltanzadeh et al. [15] also support this research by highlighting the significance of strategic placement of vertical access features in high-rise buildings, which can potentially reduce total evacuation time by 20%. Another approach to improve evacuation efficiency is by having wider and more spacious corridors. For evacuation on staircases, research shows that ascending pedestrians can move faster than descending ones with the existence of stair-landings, which can significantly improve unidirectional evacuation [16]. Xie et al. [17] also conducted research on staircase evacuation, where their research focused on using smart technology to predict evacuee density to help provide dynamic evacuation paths.

Building upon this foundation, Chen et al. [18] proposed a dynamic evacuation simulation model based on cellular automata, considering pedestrian movement and collision avoidance behaviour. Lizhong et al. [19] extended this concept, incorporating building characteristics to understand unique scenarios during evacuation. Both studies stress the importance of comprehending human behaviour and building layouts during emergencies. Balboa et al. [20] findings also indicate that integrating human behaviour and smart building technologies has the potential to enhance the evacuation process.

However, nowadays, research seems to focus on the development of agent-based model simulation for the evacuation process. An agent-based model is an evacuation modelling approach designed to combine human thinking and the decision-making process into an evacuation process simulation [21]. Agent-based models also present people as independently intelligent and autonomous agents that have the ability to sense, interact and adapt to dynamic environments [22]. There are three key concepts of an agent-based model that can be extracted from the previous descriptions: intelligent agent, model and simulation. An intelligent agent can be defined as a virtual or physical entity that can act autonomously to interact, learn and adapt to its environment [23]. A model can be defined as a mathematical, computerised and graphical representation of objects and their relationship in an environment. Simulation is a method to understand the functionality of a complex system without involving actual real-life situations. Despite the growing trend, agent-based technologies are unable to model all evacuation factors and their interactions simultaneously. Therefore, it is essential to review prior research to identify additional evacuation factors and potential new areas of interest in evacuation studies.

One of the prior studies focused on smart buildings was conducted by Wehbe et al. [24]. This study focused on smart buildings, presenting a system integrating building information modelling (BIM) and smart technologies for early fire detection and optimal evacuation path identification. Similarly, Gokceli et al. [25] proposed a building automation system that can manage various hardware and control services in emergency scenarios that incorporate IoT-based design, proving that smart building technology is useful during the evacuation process.

The other method to get an accurate evacuation process model is by developing different evacuation models needed to support different evacuation backgrounds [26]. However, this practice is uneconomical and hardly possible. So, one approach that has become a growing trend because of its superior properties in model design is metamodel [27]. This is because metamodel is an abstract model that defines a model in which an abstract model is developed using only allowed symbols that conform to modelling language rules to describe a domain model formally. So, instead of requesting a unique model built for each evacuation background, the development of a metamodel that describes the permitted structure that all evacuation models must comply with seems more appropriate.

Despite the advancement in existing studies and the mentioned advantages of metamodel development, a notable gap exists in considering the potential advantages of the metamodel concept for developing a flexible and scalable evacuation model. This research aims to fill this gap by introducing a metamodel that addresses the limitations observed in previous studies.

3. DEVELOPMENT METHOD

In this research, a safety evacuation metamodel for smart buildings is proposed. To develop the proposed metamodel, a systematic literature review was done, resulting in a total of 76 research papers exhaustively explored. All research was analysed, and the features were extracted, making sure that all evacuation factors were considered. The following section will describe the method used to develop the safety evacuation metamodel. In this research, the Eclipse Modelling Framework (EMF) is used to support the development of the proposed metamodel. EMF is chosen as the medium to design the metamodel because of its ability to support various platforms for language definition and its ease of use in describing graphical models [28] [29]. The proposed base metamodel is graphically presented as a UML class diagram. This class diagram represents generic abstractions of model operations and annotation models as initially proposed by De Lara et al. [30]. This research develops the metamodel in two stages. The first stage is to develop the initial metamodel that is derived from several selected research. The purpose of this initial concept is to provide a draft for a set of generic concepts useful for modelling smart evacuation language, while the second stage is to verify the metamodel component presented in the initial metamodel and support the improvement of the metamodel.

3.1 Initial Metamodel

The initial metamodel was developed through a systematic literature review encompassing 35 model features. The systematic literature review focuses on the growing interest in evacuation metamodels from 2015 to 2022. This period is selected to avoid bias from growing interests and rapidly evolving concepts in evacuation modelling. Also, this period helps to provide more stable and insightful literature to the research.

The term “building” has emerged as the most frequently referenced term, with Kai et al. [31] utilising “building setting” and Soltanzadeh et al. [32] employing the term “building” in vertical evacuation research. Defined as an enclosed structural model providing spatial information, this led to the exclusion of certain features like doors, elevators, roofs, sites, and slabs from the resulting model features [33]. Additionally, terms such as “data model”, “environment condition”, “evacuee agent”, and “obstacles” were recurrent in the selected review research. “Data model” was defined as providing relevant building and occupant traffic information by [34]. “Data model” is also defined as a model that describes the properties and relationship of data sources [35]. Considering these definitions, the “human evacuation” term can be considered as an example of a “data model”, resulting in its exclusion from the initial metamodel features. “Environmental condition” mentioned in Kasereka et al. [36] evacuation model referred to spatial and demographic information, with “spatial analysis” being illustrative features [37]. “Evacuee agents” are people who have their own behaviour and features where they can see fire, hear emergency alarms and evacuate from the building [38]. “Evacuee agent” is described as an individual with behaviours similar to “occupant”, leading to the latter’s exclusion from the initial metamodel. The preference for the term “evacuee” over “occupant” was due to its higher frequency in the literature.

“Obstacle” was characterised as a physically blocked area within a building [39]. Seven terms were mentioned twice in the resulting literature: “Alarm”, “Emergency utilities”, “Fire”, “Hazard”, “Plan agent”, “Signage”, and “Smoke”. For instance, “alarm” refers to a fire alarm application detecting fire within a range, while “emergency utilities” denote objects used in firefighting [40]. Surprisingly, “fire” was not as frequently mentioned as expected despite the focus of the study being fire evacuation, potentially due to “hazard” being used interchangeably [41]. “Smoke” consists of toxic substances that are bad for evacuee health and one of the major causes of fatality in fire emergency scenarios [42]. Since “smoke” is a part of a fire feature, because “smoke” cannot exist on its own, “smoke” and “fire” are replaced by the term “hazard”. Meanwhile, the “plan agent” is the feature that helps guide evacuees to exit by providing escape routes and directions, similar to “signage” that provides direction for evacuees.

All the mentioned terms and features are considered as main components of the proposed metamodel. While other model features mentioned only once were excluded from the initial metamodel development. This is because these features are very specific and unique to their respective particular study only, making them less likely to represent an essential component of this study.

3.2 Verification Metamodel

The focus of this stage is the refinement of the smart evacuation metamodel to ensure its completeness. The verification process in this stage crosschecks a set list derived from the literature review to assess how well the proposed metamodel accommodates smart evacuation concepts. The literature review study identifies an additional 35 model features to be included in the proposed metamodel.

After the verification and enhancement of the initial metamodel, a total of 12 components are designed for the proposed metamodel. Some components are removed, such as “alarm” and “signage”, while others, including “network service”, “smart devices”, “actuators”, and “sensors”, are added based on additional model features in this stage. The rationale for removing alarms and signage lies in the abstraction of these concepts into the broader category of actuators within the smart building context. The added model features were considered because the smart building is a part of cyber-physical systems, consisting of physically embedded devices and network services. The term “sensor detection” and the term “fire sensor detection” refer to the ability of smart buildings to detect fire with the help of sensors [43]. The term “actuator” is used to refer to devices that help in smart building functionality [44]. For instance, the terms “fire utilities” and “emergency utilities” were acknowledged as a part of actuators, making them removed from the proposed metamodel. The need for network connections in smart devices highlights the significance of the term “network transmission” for facilitating communication and information exchange between data and planning systems, as well as among smart devices and planning systems [45].

Based on the abstraction and detailed information on all reviewed components, some components can lead to additional classes from the initial metamodel, and some can lead to added class attributes. For instance, the terms “efficient path”, “pedestrian distribution”, “fire situation awareness”, and “risk calculation” are integrated into the planning system class. Additionally, “location” attributes are introduced for classes such as obstacle, hazard, and evacuee. Attributes like “exit” and “area” are incorporated into the building class. However, there is a query regarding the inclusion of “area” and “geometry for the building classes, given that evacuees’ primary concern is finding the exit. The distinction is made by considering “geometry” as a part of the “obstacle” component, making a separate definition unnecessary. Another aspect to be considered from this stage is the evacuee attributes. evacuee attributes incorporate features such as “density”, “behaviour”, “characteristics”, “knowledge”, “performance”, “location”, “distance” and exit knowledge” as mention in the study that focuses more on the evacuee behaviour during evacuation process [46], [47].

Certain model features from the verification metamodel literature list are omitted due to several factors, such as overlapping with existing features or the features having a different focus from the smart evacuation system. The final metamodel is proposed and developed through thoughtful consideration and refinement, taking into account the verification results and additional model features identified in the literature.

4. THE PROPOSED METAMODEL

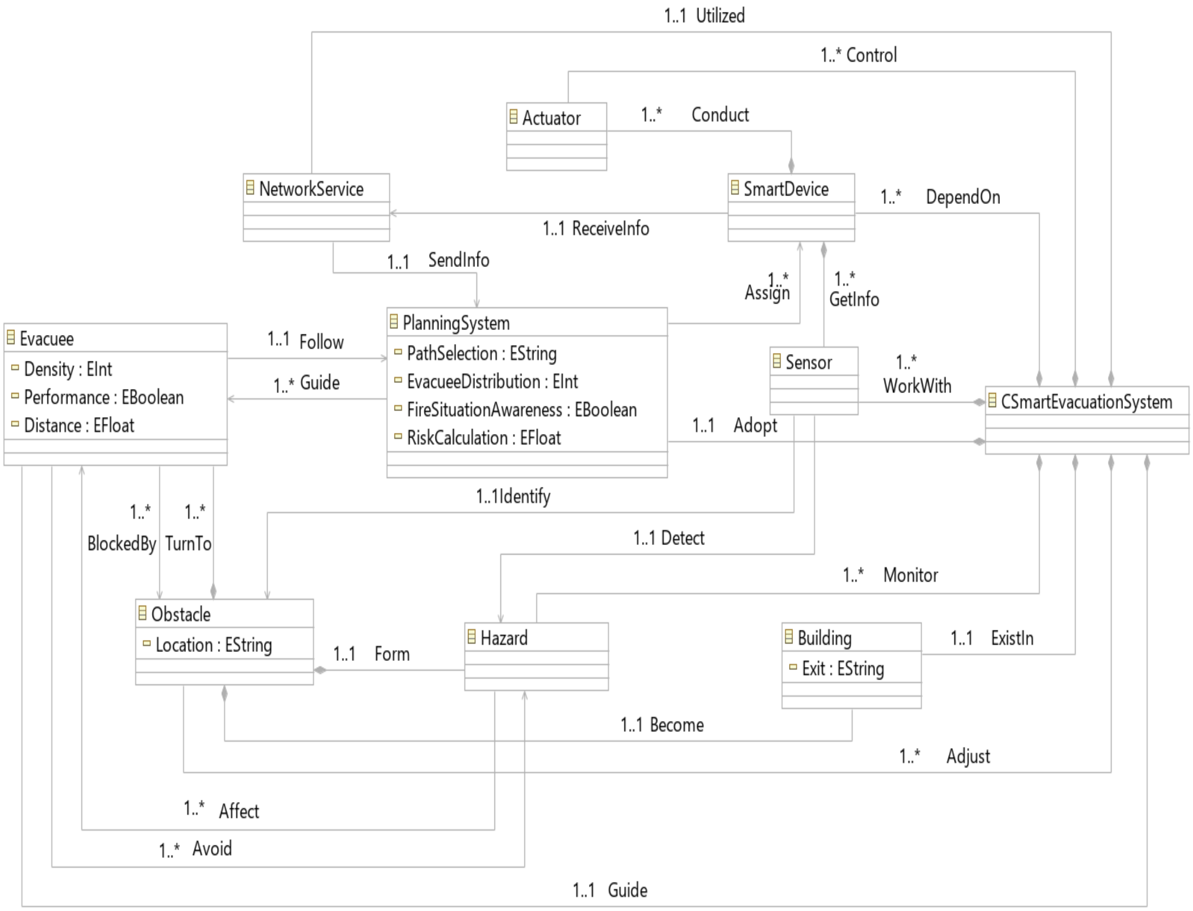


Fig. 1: Safety evacuation metamodel

The final version of the proposed metamodel, as depicted in Fig. 1, differs from the initial metamodel version. Fig. 1 shows the ten main concepts of the safety evacuation metamodel. These concepts of the proposed metamodel are described as follows:

C Smart Evacuation System: C Smart Evacuation System is the root class of the metamodel that manages, controls, and holds all other classes together.

Planning System: The planning system represents the system that collects and analyses data from the emergency environment condition.

Path Selection: Path selection is a feature of the planning system where a path is selected, and the evacuee is guided to this path based on the current environmental condition

Evacuee Distribution: Evacuee distribution is a feature that distributes evacuees evenly to every exit available, reducing overcrowding in one exit.

Fire Situation Awareness: Fire situation awareness is the system's acknowledgement of the current status of the fire emergency.

Risk Calculation: Risk calculation works collaboratively with fire situation awareness to identify the risk of each evacuation element based on the suggested evacuation path.

Smart Device: Smart devices are a composite of sensors and actuators.

Sensor: Sensors are devices that detect and receive information from hazards and obstacles.

Actuator: Actuators are devices that operate on the information received from the sensor.

Network Service: Network service is important to communicate and to allow information exchange between data and the planning system as well as between smart devices and the planning system

Building: The building represents the layout of the emergency building. Based on the layout, this feature can become an obstacle or additional help to the evacuation process.

Exit: Exits represent the destination of all evacuees in the evacuation process.

Obstacle: Obstacles are features that can slow down evacuee movement and increase the evacuation time, thus exposing evacuees to more threats.

Location: Location concerns the placing of the obstacle.

Hazard: A hazard is defined as a threat that can affect evacuees and building structures. Examples of hazards are fire, smoke, and toxic gases.

Evacuee: Evacuee is a person who is trapped in an emergency building.

Density: Density is an attribute of evacuees that concerns the number of evacuees present in the evacuation building

Performance: Evacuee performance is defined as how evacuees perform evacuation action based on their behaviour, character, and knowledge

Distance: Distance concerns the length between the evacuee's position and the exit location.

These metamodel features can be extended and added to adapt to any building environment.

5. MODEL CHECKING VALIDATION

Model checking is selected as the evaluation technique because it is one of the most useful techniques to verify safety and security protocols as well as validate the correctness and completeness of a model [48]. Research by Y.Y. Nazaruddin et al. also shows that the model-checking technique using the UPPAAL model checker effectively verifies the safety of a software system, achieving similar results as simulations [49]. The next section will introduce the case study used in this research, followed by the evaluation process, and lastly, the evaluation results are discussed.

5.1 Evacuation Case Study

This evaluation process is conducted based on a smart residential building that offers a simple building layout with multiple exits along with a diverse set of possible evacuee characteristics and behaviours. The scenario simulates a fire emergency scenario on the third floor of the five-floor residential building.

The first floor consists of two main exits and four emergency exits, while the other floors consist of four emergency exits that are connected to the first floor by emergency stairs. This residential building also has an elevator system that can be used to transport occupants to the first floor. However, the use of elevators during a fire emergency is not advisable. This is because it can cause severe congestion. This elevator will only be used for disabled occupants. The front structure of this building is shown in Fig. 2. In this case study, a total of 250 occupants are involved in the fire emergency. Each floor has an equal number of occupants, 50 occupants. Each occupant will be directed to an emergency exit or main exit. These exist and have different capabilities, with emergency exits that can accommodate 30 occupants at a time, and the main exists that can handle up to 60 occupants simultaneously. In this emergency scenario, fire propagates slowly, which provides enough time for the firefighter to arrive at the location before the whole floor is affected. However, all occupant is required to evacuate the building to avoid fatal damage to the occupant.

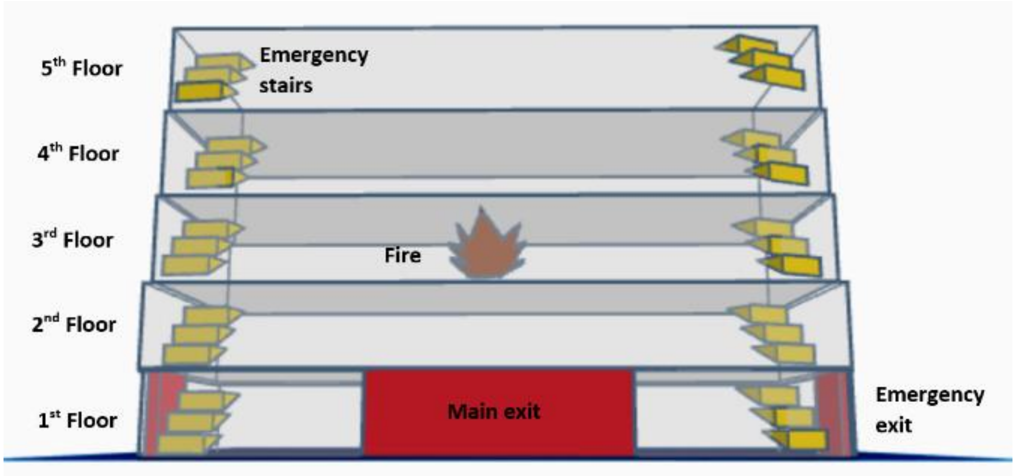


Fig. 2. Building structure

5.2 Model transformation

Model transformation is a crucial step in many modelling research in automating the generation of target models from the source models. In this research, the UPPAAL model checker is used to evaluate the metamodel. However, as the proposed metamodel is presented in the class diagram, it has become a limitation to the evaluation process because the UPPAAL model checker is only capable of handling state machine diagrams. To overcome this limitation, a powerful and versatile Atlas Transformation Language (ATL) was employed to transform the metamodel seamlessly. The transformation process follows a structured pattern where the source model must conform to the source metamodel, while the transformed target model must conform to the target metamodel. The ATL program also needs to follow the ATL metamodel [50] strictly. The ATL rules and structured pattern is displayed in Fig. 3.

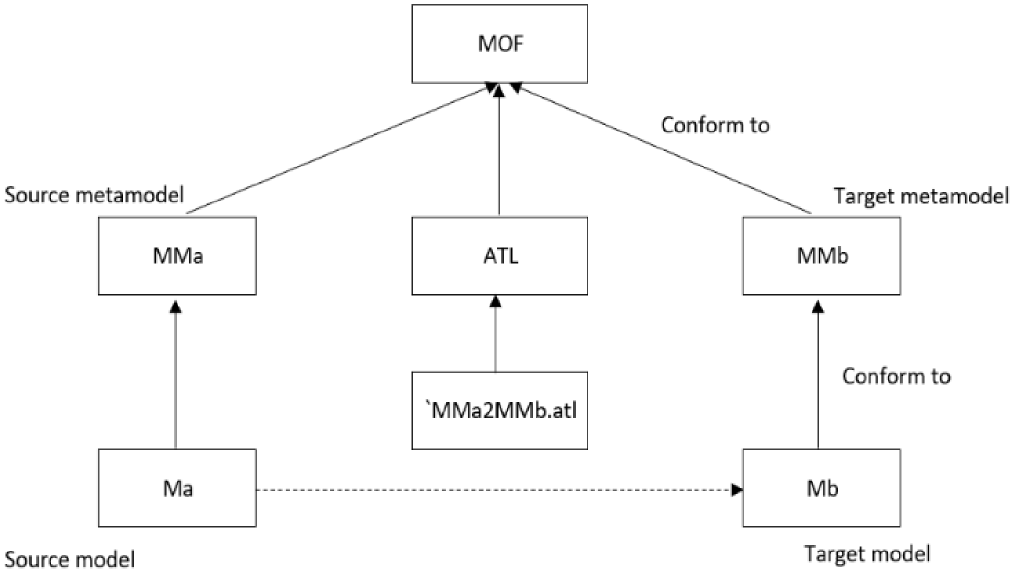


Fig. 3: ATL pattern

ATL can be composed into three parts: header, helpers, and rules [51]. The effectiveness of the transformation process relies on ATL helpers, which consist of operation helpers and attribute helpers. Operation helpers facilitate navigation over the source model that can accept input parameters, whereas attribute helpers link the source model element to any read-only value. These ATL helpers are crucial during the model transformation process to guarantee precise mapping and accurate transformation elements.

The ATL execution engine operates based on a Virtual Machine (VM) that executes bytecode, making it necessary to compile the ATL code into bytecode. Notably, Eclipse EMF can support ATL execution, providing seamless integration within the EMF platform. To promote efficient and streamlined model transformation, this research uses EMF to enable the automatic execution of the model transformation process.

There are three main steps in the model transformation process for this research. Firstly, the creation of the source metamodel and the target metamodel as a basis of the model transformation. Secondly, the creation of the source model follows strictly the source metamodel. Lastly, to ensure precise mapping and transformation, the transformation rules are defined. After completing all three steps and conforming to precise metamodel transformation, the model-checking evaluation process can take place.

5.3 Evaluation process

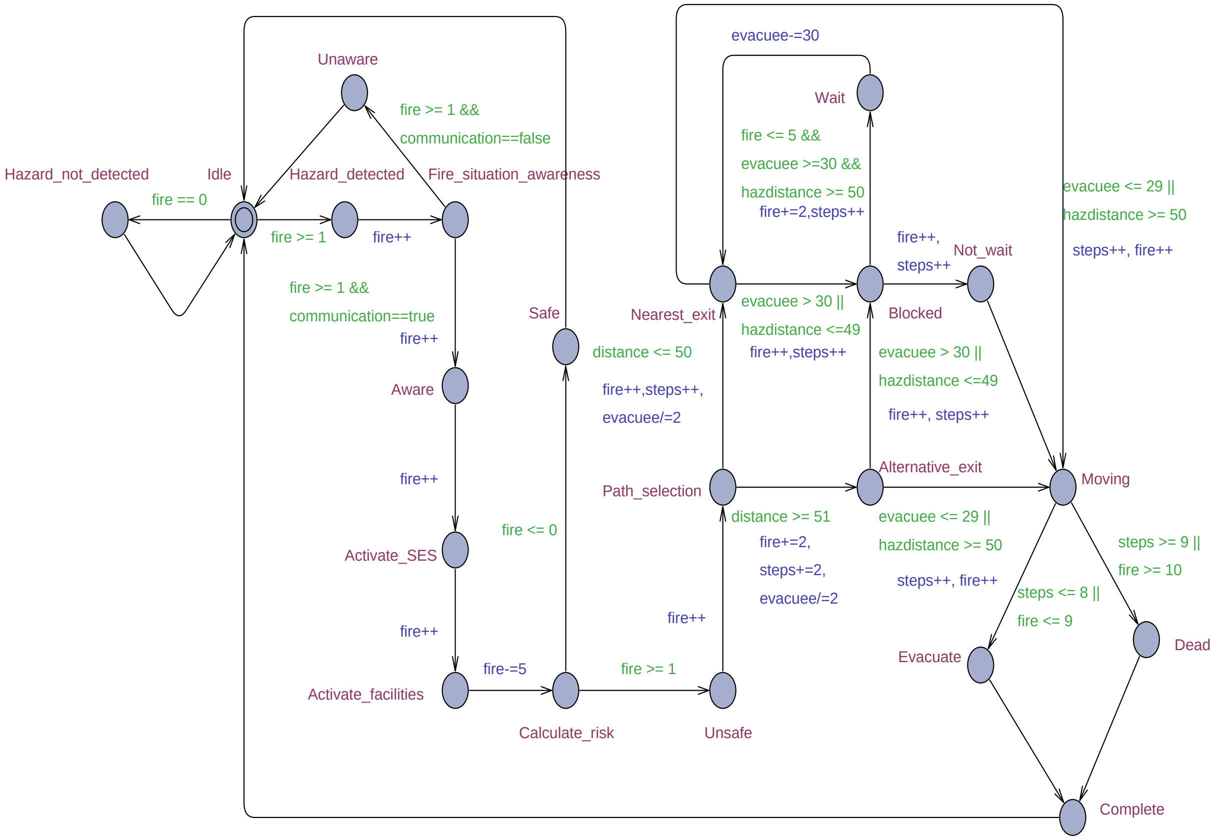


Fig. 4: Evacuation process flow

UPPAAL model checker uses timed automata and the UPPAAL toolbox to predict, analyse, simulate and verify the properties in a software system [52]. Based on the method of evaluation by the UPPAAL model checker, along with the case study data and model transformation result, metamodel properties can be evaluated based on the model display in Fig. 4. These properties will represent each characteristic of the safety evacuation metamodel. The abstraction of the metamodel makes it possible to verify the properties of this extensive safety metamodel design automatically using the UPPAAL model checker [53].

The first property is to check the actuator class. This feature can be checked by having an actuator action in the smart evacuation system where the actuator will be able to reduce the fire in an emergency. This property means that ‘There should be a situation in which emergency facilities are activated where fire hazard is detected'. A return true value is expected from this property.

Query 1. *E<> (P.Activate_facilities && P.fire>=1)*

The second property to be verified is the sensor class. This property will ensure that the functionality of the sensor is properly declared and checked. The following property describes that 'There exists a situation where sensor detect hazard when the hazard does not exist in the building'. A true return value is expected from this property. This is because the sensor will be able to send detection information when a hazard is present.

Query 2. *E<> (P.Hazard_detected && P.fire>=1)*

The next property to be verified is network services. These metamodel features can be verified by ensuring that a transition is done when the system receives a communication request. This property can be described as 'There should not exist a situation where the system is unaware of the fire situation if the communication is reachable'. A true value is expected from this property, representing that the system can perform transition because of communication achieved between each smart evacuation system component.

Query 3. *E<> not (P.Unaware && P.communication==true)*

Another property to be verified is the evacuee. The first evacuee property to be checked is density. This is checked by ensuring that the exit is blocked when maximum evacuee density is reached. The following property means that 'The exit can be blocked when the evacuee density at the exit is less than 30 evacuees.' A false value is expected from this property.

Query 4. *E<> (P.Blocked && P.evacuee <= 30)*

The next property is evacuee performance. This performance is described as the evacuee movement performance in a fire emergency. To check this property, the steps count taken by the evacuee is measured. The result of evacuee performance can be divided into successfully evacuated or dead. This property shows that 'There can exist a situation where all evacuees can successfully evacuate from the building'. This condition is expected to be true.

Query 5. *E<>P. Evacuate*

The final evacuee property to be checked is distance. The evacuee should be directed to the nearest exit. Thus, the property should check that the evacuee is moved to an exit if their distance is less than 51 meters from the exit. This property should return true value, indicating that a situation where the evacuee is directed toward an alternative exit should never exist if the evacuee's distance is less than or equal to 50 meters.

Query 6. *E<> (P.Nearest_exit && P.distance<=50) && not (P.Alternative_exit && P.distance<=50)*

Following the model-checking process, the planning system is next to be checked. This can be done by verifying the event conducted by the evacuation system. This event includes path selection, evacuee distribution, fire situation awareness, and risk calculation. By using the same Query 6 checking, path selection can be indirectly checked.

Fire situation awareness can be checked by ensuring that network services receive information when sensors detect a hazard. This property cannot be checked individually; instead, this property will depend on the action taken by network services. So, properly checking those properties will indirectly check for fire situation awareness properties. However, evacuee distribution can be checked by defining property as follows:

Query 7. *E<> (P.Nearest_exit && P.evacuee) == (P.Alternative_exit && P.evacuee)*

This property shows that 'There exists a situation where evacuee density at nearest exit is the same as evacuee density at the alternative exit.' This property is expected to return true value.

Next, risk calculation is important to determine system action during an emergency. 'For every situation, the system will consider the building as unsafe if the fire risk exists.' So, the true return value is expected from the following defined property.

Query 8. *A<> (P.Unsafe && P.fire >= 1)*

Meanwhile, obstacle features can be checked when the condition of an obstacle is met. For instance, the property will check if 'The path is blocked when evacuee distance from a hazard is more than 50 meters.' This statement is not true, so a false value is expected from this property checking.

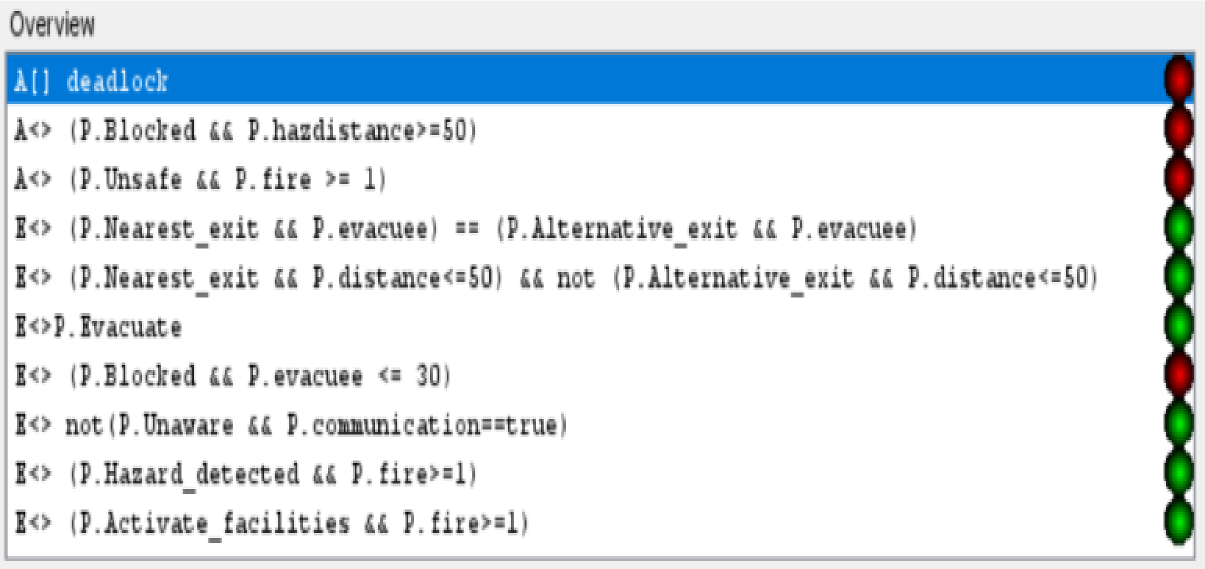
Query 9. *E<> (P.Blocked && P.hazdistance >= 50)*

The smart device is another property that can be checked indirectly. Since sensors and actuators are examples of smart devices, by correctly defining and functioning sensors and actuators in a smart evacuation system, smart devices are considered properly checked.

The next property that can be checked is whether the system can work correctly. This can be checked with the property as follows:

Query 10. *A[] deadlock*

A deadlock means that the system cannot work and crash. This should not happen; thus, the property checking should return a false value.



```
A[] deadlock
A<> (P.Blocked && P.hazdistance>=50)
A<> (P.Unsafe && P.fire >= 1)
E<> (P.Nearest_exit && P.evacuee) == (P.Alternative_exit && P.evacuee)
E<> (P.Nearest_exit && P.distance<=50) && not (P.Alternative_exit && P.distance<=50)
E<>P.Evacuate
E<> (P.Blocked && P.evacuee <= 30)
E<> not(P.Unaware && P.communication==true)
E<> (P.Hazard_detected && P.fire>=1)
E<> (P.Activate_facilities && P.fire>=1)
```

Fig. 5: Model checking result

The expected result and model checking result are compared to identify possible errors that exist from this verification. These results are shown in Table 1.

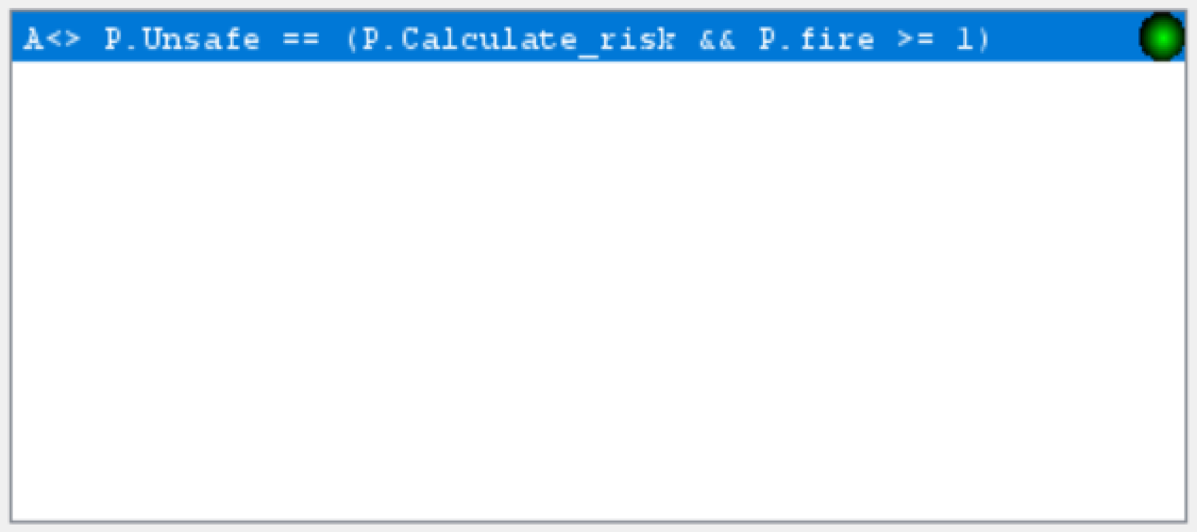
Table 1: Result Comparison

Query	Expected results	Model checking results
1	True	True
2	True	True
3	True	True
4	False	False
5	True	True
6	True	True
7	True	True
8	True	False
9	False	False
10	False	False

Referring to the comparison table, all queries meet the expected result except for Query 8. Query 8 is defined to check whether the system can correctly determine the status of the building based on the presence of a hazard. Therefore, the system should always consider the smart building status as unsafe if a hazard is present. However, the result of Query 8 shows that the system does not always define the building as unsafe even though a hazard is present.

After fully analysing the condition and UPPAAL model, this property cannot be satisfied because the system considers the fire count from the start of the smart evacuation system instead of after the planning system is activated. By doing so, the calculated risk state cannot be reached. Thus, this property can be rechecked by assuming that the planning system is successfully activated. A new query, Query 11, is defined to check the system's correctness in determining the building status based on the risk calculated.

Query 11. *A<> P.Unsafe == (P.Calculate_risk && P.fire >= 1)*



```
A<> P.Unsafe == (P.Calculate_risk && P.fire >= 1)
```

Fig. 6: Result of Query 11

Based on Fig. 6, this query returns a true value showing that the smart evacuation system can correctly determine the building status based on the risk calculation when the planning system is properly activated. Although the system can remain idle, if the hazard is present, the sensor will be able to detect the hazard. This is confirmed by the model checking results where the sensor property is correctly working.

6. CONCLUSION

This paper proposed a safety evacuation metamodel designed to optimise the evacuation process in smart building systems. The proposed metamodel aims to bridge the gap between traditional fire evacuation models and the advancement of smart building technologies, incorporating real-time data, smart devices, and more advanced technologies. The metamodel was developed based on various evacuation terms that act as grammar guidelines. The integration of metamodel concepts and model transformation techniques ensures that the evaluation process is rationalised. Through a case study of a smart five-floor residential building and model checking evaluation, the metamodel's properties are systematically verified, ensuring the correctness and effectiveness of each component of the proposed metamodel in the evacuation process. This research contributes to a significant advancement in the field of smart evacuation planning by providing a comprehensive and adaptable framework to improve safety measures within smart buildings during fire emergencies. However, this metamodel cannot fully represent evacuee behaviour during the evacuation process. In developing an abstract and flexible metamodel, the inclusion of evacuee characteristics and behaviour is not possible. Future work might take advantage of this gap to develop an evacuation metamodel that focuses on the different evacuee characteristics and behaviours.

References

1) Le, A. T. H., Park, K. S., Domingo, N., Rasheed, E., & Mithraratne, N. (2021). Sustainable refurbishment for school buildings: a literature review. International Journal of Building Pathology and Adaptation, 39(1), 5-19.

2) Tan, S., Weinert, D., Joseph, P., & Moinuddin, K. A. (2021). Incorporation of technical, human and organisational risks in a dynamic probabilistic fire risk model for high-rise residential buildings. Fire and materials, 45(6), 779-810.

3) Portal Rasmi Jabatan Bomba Dan Penyelamat Malaysia. (n.d.). JBPM 2021 Annual Report. Malaysian Fire and Rescue Department. https://www.bomba.gov.my/wp-content/uploads/2023/01/Laporan-Tahun-2021_compressed.pdf

4) Gan, W., Chen, C., Wang, Z., Song, J., Kuang, Y., He, S., Mi, R., Sunderland, P., & Hu, L. (2019). Dense, Self-Formed Char Layer Enables a Fire-Retardant Wood Structural Material. Advanced Functional Materials, 29.

5) Barański, M., & Haznar-Barańska, A. (2021). Evacuation and its Types – Revision of the Definition and Classification. Safety & Fire Technology.

6) Kolmann, P. (2020). Z problematyki ochrony ludności. Istota ewakuacji w ujęciu powszechnym. De Securitate et Defensione. O Bezpieczeństwie i Obronności, 6(1), 88-98. Groner, N. (2016). A Decision Model for Recommending Which Building Occupants Should Move Where During Fire Emergencies. Fire Safety Journal, 80, 20-29.

7) Groner, N. (2016). A Decision Model for Recommending Which Building Occupants Should Move Where During Fire Emergencies. Fire Safety Journal, 80, 20-29.

8) Zhang, L., Wang, J., & Shi, Q. (2014). Multi-agent based modeling and simulating for evacuation process in stadium. Journal of Systems Science and Complexity, 27, 430-444.

9) Heo, G., Kim, Y., Seo, S., Jeon, S., Lee, J., & Kim, D. (2021). Development of Smart Evacuation System for Fire Response in Apartments. Journal of the Korean Society of Hazard Mitigation.

10) Karsai, G., Nordstrom, G., Lédeczi, Á., & Sztipanovits, J. (2000). Specifying graphical modeling systems using constraint-based meta models. CACSD. Conference Proceedings. IEEE International Symposium on Computer-Aided Control System Design (Cat. No.00TH8537), 89-94.

11) Liu, T., Luo, W., & Yan, B. (2013). The Application of the Sensor Model Language in the HeiHe Watershed Allied Telemetry Experimental Research. , 486-497.

12) Kaitao, C., Quanbo, G., & Shenan, D. (2012). Research and design for emergency evacuation instructions system based on wireless sensor network. 2012 24th Chinese Control and Decision Conference (CCDC), 3849-3854.

13) Ronchi, E., & Nilsson, D. (2013). Fire evacuation in high-rise buildings: a review of human behaviour and modelling research. Fire Science Reviews, 2, 1-21.

14) Han, Z., Weng, W., Zhao, Q., Ma, X., Liu, Q., & Huang, Q. (2013). Investigation on an Integrated Evacuation Route Planning Method Based on Real-Time Data Acquisition for High-Rise Building Fire. IEEE Transactions on Intelligent Transportation Systems, 14, 782-795.

15) Soltanzadeh, A., Mazaherian, H., & Heidari, S. (2021). Optimal solutions to vertical access placement design in residential high-rise buildings based on human behavior. Journal of building engineering, 43, 102856. https://doi.org/10.1016/J.JOBE.2021.102856.

16) Gao, J., Gong, J., He, J., Zhang, D., Qiu, G., & Zhang, J. (2020). Experimental study on pedestrians' uni- and bi-directional movement on staircases under emergency conditions. Journal of Statistical Mechanics: Theory and Experiment, 2020.

17) Xie, R., Pan, Y., Zhou, T., & Ye, W. (2020). Smart safety design for fire stairways in underground space based on the ascending evacuation speed and BMI. Safety Science, 125, 104619.

18) Chen, L., Tang, T., Huang, H., Wu, J., & Song, Z. (2018). Modeling pedestrian flow accounting for collision avoidance during evacuation. Simul. Model. Pract. Theory, 82, 1-11.

19) Lizhong, Y., Weifeng, F., & Weicheng, F. (2003). Modeling Occupant Evacuation Using Cellular Automata - Effect of Human Behavior and Building Characteristics on Evacuation. Journal of Fire Sciences, 21, 227 - 240.

20) Balboa, A., González-Villa, J., Cuesta, A., Abreu, O., & Alvear, D. (2020). Testing a real-time intelligent evacuation guiding system for complex buildings. Safety Science, 132, 104970.

21) Wang, W., Lo, S., Liu, S., & Ma, J. (2015). On the use of a pedestrian simulation model with natural behavior representation in metro stations. Procedia Computer Science, 52, 137-144.

22) Edrisi, A., Lahoorpoor, B., & Lovreglio, R. (2021). Simulating metro station evacuation using three agent-based exit choice models. Case Studies on Transport Policy, 9(3), 1261-1272. doi: https://doi.org/10.1016/j.cstp.2021.06.011

23) Bonabeau, E. (2002). Agent-based modeling: Methods and techniques for simulating human systems. Proceedings of the national academy of sciences, 99(suppl_3), 7280-7287.

24) Wehbe, R., & Shahrour, I. (2021). A BIM-Based Smart System for Fire Evacuation. Future Internet, 13, 221.

25) Gökceli, S., Zhmurov, N., Karabulut-Kurt, G., & Örs, B. (2017). IoT in Action: Design and Implementation of a Building Evacuation Service. J. Comput. Networks Commun., 2017, 8595404:1-8595404:13.

26) 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.

27) Mullur, A. A., & Messac, A. (2006). Metamodeling using extended radial basis functions: a comparative approach. Engineering with Computers, 21(3), 203-217.

28) Carré, B., Vanwormhoudt, G., & Caron, O. (2018). On submodels and submetamodels with their relation. Software & Systems Modeling, 17(4), 1105-1137.

29) Durak, U., Pawletta, T., Oguztuzun, H., & Zeigler, B. P. (2017). System entity structure and model base framework in model based engineering of simulations for technical systems. Paper presented at the Proceedings of the Symposium on Model-driven Approaches for Simulation Engineering.

30) De Lara, J., Guerra, E., & Cuadrado, J. S. (2013). Reusable abstractions for modeling languages. Information Systems, 38(8), 1128-1149.

31) Kai, H., Xuewei, L., & Lei, X. (2020). Simulation of hotel fire and safe evacuation based on FDS and Pathfinder. Paper presented at the 2020 International Conference on Robots & Intelligent System (ICRIS).

32) Soltanzadeh, A., Mazaherian, H., & Heidari, S. (2021). Optimal solutions to vertical access placement design in residential high-rise buildings based on human behavior. Journal of Building Engineering, 43, 102856. doi: https://doi.org/10.1016/j.jobe.2021.102856

33) Tang, F., & Ren, A. (2008). Agent-based evacuation model incorporating fire scene and building geometry. Tsinghua Science and Technology, 13(5), 708-714.

34) Boje, C., & Li, H. (2018). Crowd simulation-based knowledge mining supporting building evacuation design. Advanced Engineering Informatics, 37, 103-118.

35) Cicirelli, F., Fortino, G., Guerrieri, A., Spezzano, G., & Vinci, A. (2017). Metamodeling of smart environments: from design to implementation. Advanced Engineering Informatics, 33, 274-284.

36) Kasereka, S., Kasoro, N., Kyamakya, K., Goufo, E.-F. D., Chokki, A. P., & Yengo, M. V. (2018). Agent-Based Modelling and Simulation for evacuation of people from a building in case of fire. Procedia Computer Science, 130, 10-17.

37) Chen, J., Ma, J., & Lo, S. (2017). Event-driven modeling of elevator assisted evacuation in ultra high-rise buildings. Simulation Modelling Practice and Theory, 74, 99-116.

38) Nguyen, M. H., Ho, T. V., & Zucker, J.-D. (2013). Integration of smoke effect and blind evacuation strategy (SEBES) within fire evacuation simulation. Simulation Modelling Practice and Theory, 36, 44-59.

39) Liu, J. W. S., Lin, F. T., Chu, E. T. H., & Zhong, J. (2016, 6-7 Dec. 2016). Intelligent indoor emergency evacuation systems: Reference architecture and data requirements. Paper presented at the 2016 Future Technologies Conference (FTC).

40) Tashakkori, H., Rajabifard, A., & Kalantari, M. (2015). A new 3D indoor/outdoor spatial model for indoor emergency response facilitation. Building and Environment, 89, 170-182.

41) Li, Z., Huang, H., Li, N., Zan, M. L. C., & Law, K. (2020). An agent-based simulator for indoor crowd evacuation considering fire impacts. Automation in Construction, 120, 103395.

42) Mei, F., Tang, F., Ling, X., & Yu, J. (2017). Evolution characteristics of fire smoke layer thickness in a mechanical ventilation tunnel with multiple point extraction. Applied Thermal Engineering, 111, 248-256.

43) Tang, J.-X., & Jia, Q.-S. (2019). A simulation platform for sensing system selection for occupant distribution estimation in smart buildings. Paper presented at the 2019 IEEE 15th International Conference on Automation Science and Engineering (CASE).

44) Majumder, S., O'Neil, S., & Kennedy, R. (2017). Smart apparatus for fire evacuation—An IoT based fire emergency monitoring and evacuation system. Paper presented at the 2017 IEEE MIT Undergraduate Research Technology Conference (URTC).

45) Xu, X., Zhang, L., Sotiriadis, S., Asimakopoulou, E., Li, M., & Bessis, N. (2018). CLOTHO: A large-scale Internet of Things-based crowd evacuation planning system for disaster management. IEEE Internet of Things Journal, 5(5), 3559-3568.

46) Niu, Y., Zhang, J., Zhang, Y., & Xiao, J. (2019). Modeling evacuation of high-rise buildings based on intelligence decision P system. Sustainability, 11(17), 4685.

47) Cao, Y., Luo, C., Liu, Y., Teng, S., & Xin, G. (2021). Path intelligent optimisation for dense crowd emergency evacuation in heritage buildings. Journal of Cultural Heritage, 47, 180-187. doi: https://doi.org/10.1016/j.culher.2020.06.007

48) Ma, J., Zhang, D., Xu, G., & Yang, Y. (2010). Model Checking Based Security Policy Verification and Validation. 2010 2nd International Workshop on Intelligent Systems and Applications, 1-4.

49) Nazaruddin, Y., Tamba, T., Pradityo, K., Aristyo, B., & Widyotriatmo, A. (2019). Safety Verification of a Train Interlocking Timed Automaton Model.IFAC-PapersOnLine.https://doi.org/10.1016/j.ifacol.2019.11.696.

50) Jouault, F., Allilaire, F., Bézivin, J., Kurtev, I., & Valduriez, P. (2006, October). ATL: a QVT-like transformation language. In Companion to the 21st ACM SIGPLAN symposium on Object-oriented programming systems, languages, and applications (pp. 719-720).

51) Allilaire, F., Bézivin, J., Jouault, F., & Kurtev, I. (2006, July). ATL-eclipse support for model transformation. In Proceedings of the Eclipse Technology eXchange workshop (eTX) at the ECOOP 2006 Conference, Nantes, France (Vol. 66). Citeseer.

52) Mouelhi, S., Laarouchi, M., Cancila, D., & Chaouchi, H. (2019). Predictive Formal Analysis of Resilience in Cyber-Physical Systems. IEEE Access, 7, 33741-33758. https://doi.org/10.1109/ACCESS.2019.2903153.

53) Kunnappilly, A., Backeman, P., & Seceleanu, C. (2021). From UML Modeling to UPPAAL Model checking of 5G Dynamic Service Orchestration. 7th Conference on the Engineering of Computer Based Systems.