



HAL
open science

iTEVAC: an enhanced trusted evacuation system leveraging fog computing and IoT

ThinH Vinh Le, Huan Thien Tran, Samia Bouzefrane

► To cite this version:

ThinH Vinh Le, Huan Thien Tran, Samia Bouzefrane. iTEVAC: an enhanced trusted evacuation system leveraging fog computing and IoT. *Journal of Information and Telecommunication*, In press, 10.1080/24751839.2024.2306787. hal-04499928

HAL Id: hal-04499928

<https://cnam.hal.science/hal-04499928>

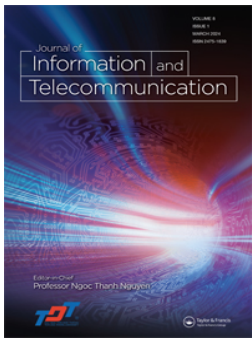
Submitted on 11 Mar 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution - NonCommercial 4.0 International License



iTEVAC: an enhanced trusted evacuation system leveraging fog computing and IoT

Thinh Vinh Le, Huan Thien Tran & Samia Bouzefrane

To cite this article: Thinh Vinh Le, Huan Thien Tran & Samia Bouzefrane (23 Jan 2024): iTEVAC: an enhanced trusted evacuation system leveraging fog computing and IoT, Journal of Information and Telecommunication, DOI: [10.1080/24751839.2024.2306787](https://doi.org/10.1080/24751839.2024.2306787)

To link to this article: <https://doi.org/10.1080/24751839.2024.2306787>



© 2024 The Author(s). Published by Informa UK Limited, trading as Taylor & Francis Group



Published online: 23 Jan 2024.



Submit your article to this journal [↗](#)



Article views: 197



View related articles [↗](#)



View Crossmark data [↗](#)

iTEVAC: an enhanced trusted evacuation system leveraging fog computing and IoT

Thinh Vinh Le ^a, Huan Thien Tran^b and Samia Bouzefrane^c

^aHo Chi Minh City of Technology and Education (HCMUTE), Ho Chi Minh City, Vietnam; ^bFaculty of Electronics and Telecommunication (FET), Saigon University (SGU), Ho Chi Minh City, Vietnam; ^cCNAM, Paris, France

ABSTRACT

This paper introduces iTEVAC, an advanced model of the Trusted Evacuation System (TEVAC), designed to optimize emergency evacuation procedures within commercial centers. Building upon the principles of TEVAC, iTEVAC incorporates additional features such as isolation, firefighting, and rescue intervention strategies to enhance the system's performance and efficiency. iTEVAC leverages the power of Fog computing and Internet of Things (IoT) devices to provide real-time guidance for individuals during emergencies. The system operates by executing alerts for isolated nodes and those unable to find an exit route, providing detailed guidance for nodes that need to move within the commercial center's network. A key advantage of iTEVAC is its decision-making speed, which is less than 1 second, crucial in high-stakes, time-sensitive situations. The iTEVAC model also addresses the limitations of older buildings, traditional markets, and crowded areas that often lack efficient and integrated evacuation systems. It allows users to rely not only on smart navigation guide panel but also on their smartphones to find the safest direction using local Wi-Fi and broadband cellular networks, supported by Fog computing. The iTEVAC model represents a significant advancement in emergency response strategies, contributing to the development of more effective and intelligent evacuation strategies.

ARTICLE HISTORY

Received 10 October 2023
Accepted 13 January 2024

KEYWORDS

Dijkstra; evacuation;
optimization; trusted path;
IoT

1. Introduction

In the face of emergencies, efficient evacuation procedures are critical to minimizing potential harm and ensuring the safety of individuals (Araujo Lima et al., 2021). Traditional evacuation methods often fall short in large commercial centres, older buildings, and crowded areas, where complex layouts and high population densities can lead to confusion and inefficiency (Cheng et al., 2016; N. Li, Yang, et al., 2014; Lurz et al., 2017; Nowak et al., 2017). In the era of rapidly evolving building architectures, the safety of occupants has emerged as a non-negotiable priority. While aesthetic and functional

CONTACT Thinh Vinh Le  thinhlv@hcmute.edu.vn  Ho Chi Minh City of Technology and Education (HCMUTE), 01 Vo Van Ngan street, Ho Chi Minh City, Vietnam

© 2024 The Author(s). Published by Informa UK Limited, trading as Taylor & Francis Group
This is an Open Access article distributed under the terms of the Creative Commons Attribution-NonCommercial License (<http://creativecommons.org/licenses/by-nc/4.0/>), which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited. The terms on which this article has been published allow the posting of the Accepted Manuscript in a repository by the author(s) or with their consent.

design elements are important, the criticality of effective fire escape planning cannot be overstated. Dynamic fire escape path planning, especially when integrated with Building Information Modeling (BIM), has proven to be indispensable in safeguarding lives during fire emergencies (Wang et al., 2021). BIM technologies have significantly advanced the field of emergency management, offering a multi-dimensional approach that includes not only 3D spatial data but also time (4D) and cost (5D), thereby providing a comprehensive framework for risk mitigation (Sloot et al., 2019). These technologies have been successfully applied to manage occupational hazards (Cortés-Pérez et al., 2020) and have been specifically adapted for the unique challenges posed by metro station evacuations (Tang et al., 2021). Furthermore, the integration of user behaviour into BIM-based fire emergency management adds another layer of complexity and effectiveness to these systems (Ma & Wu, 2020). These tools work in synergy with a variety of technologies to elevate safety measures. The fusion of BIM with indoor localization algorithms has ushered in a new era of real-time emergency response capabilities (N. Li, Becerik-Gerber, et al., 2014). These systems are further enhanced by the integration of wireless electric appliance control, offering a holistic approach to building safety (Rashid et al., 2019). Innovations in this space include performance-based evaluation methods for building evacuations, automated exit sign directions, and emergency rescue routing, all of which are centred around BIM technologies (Chen & Chu, 2016; Fu & Liu, 2020; Mirahadi et al., 2019). Moreover, The advent of smart monitoring techniques accessible via mobile applications has made the evacuation process more streamlined and effective (Cheng et al., 2017). While BIM serves as a robust framework for building safety, the incorporation of the Internet of Things (IoT) and Artificial Intelligence (AI) has added a new dimension to early fire detection and emergency response. IoT-based systems have been developed to monitor key environmental parameters such as temperature, smoke, and carbon dioxide levels, providing early warning systems that can save lives (Bhawesh & Gouthamkumar, 2020; Mehra & Tanmay, 2017; Okokpujie et al., n.d.). These technologies also enable real-time indoor tracking of occupants, thereby aiding in effective evacuation (Atila et al., 2018). Furthermore, IoT systems have been designed to control critical building infrastructure, such as power supplies and sprinkler systems, during emergencies (Al Sher-eiqi & Sohail, 2020; Sahid & Alaydrus, 2020). AI algorithms have been developed to predict fire spread based on real-time data, thereby enabling the rapid identification of safe evacuation routes fire (B. Li et al., 2018; Wu et al., 2022). Besides, the combination of the Internet of Things (IoT) and Fog computing has opened new possibilities for enhancing the safety and efficiency of emergency evacuation procedures. Generally, fog computing operates within a three-tiered (IoT Device-Fog-Cloud) platform designed to serve proximate users, offering performance similar to a cloud but with enhanced efficiency (Mahmud et al., 2018; Yousefpour et al., 2019)

The Trusted Evacuation System (TEVAC) (Vinh et al., 2023) was a significant step forward in this regard, leveraging Fog computing and IoT devices to provide real-time guidance during emergencies. In perilous and dangerous situations, relying on Cloud systems sometimes is not suitable, as the building's or area's system can be instantly disrupted when an incident occurs. Fog computing, with the advantage of being closely connected to the user, provides evacuation information within a precious amount of time. In Fog computing, the nodes act as small workstations, capable of collecting and processing information. Information about temperature, humidity, pressure, C2O concentration, as

well as the distance between any two nodes in the network is collected by an IoT node. Besides collecting this information, the IoT node also acts as an electronic cloud (Fog) helping in the transmission and processing of incident information. That is, the information at the connections between any two nodes in the network is sent back to the server, and the server acts as the Fog Master (Figure 1). TEVAC will be installed at the Fog Master.

However, while TEVAC has proven effective, there are still challenges to be addressed. This paper introduces iTEVAC, an enhanced model of TEVAC, with the following proposals:

- Reduce the time in searching for the safest route at nodes in the network.
- Improve rescue strategies when incidents occur.

The details of these two versions are illustrated in Table 1 as follows.

This table highlights the advancements and additional features of iTEVAC compared to its predecessor, TEVAC, emphasizing its improved efficiency and response in emergency situations.

The rest of this paper searches to find a comprehensive exploration of iTEVAC, the Enhanced Trusted Evacuation System leveraging Fog Computing and IoT. It is divided into five additional sections that provide a detailed understanding of the system's architecture, its related work, and the experimental work conducted to validate its efficiency. In Section 2, we critically examine existing evacuation systems and discuss their strengths and limitations. By analyzing prior research, we highlight the innovative aspects of iTEVAC that set it apart from conventional evacuation strategies. Section 3 introduces iTEVAC as a significant advancement in emergency evacuation systems, with enhanced decision-making speed and performance. The subsequent subsections delve into the architectural details and experimental validation of the system, highlighting its potential to revolutionize emergency response strategies. Then, in Section 4, we present the empirical findings obtained from extensive simulations and real-world testing of iTEVAC.

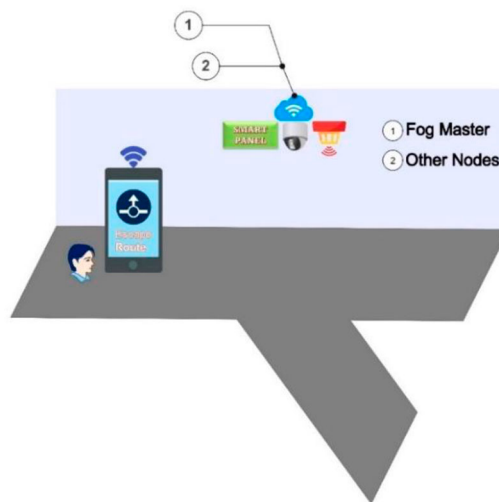


Figure 1. Fog node in TEVAC.

Table 1. Additional features of iTEVAC.

Feature	TEVAC	iTEVAC
System incorporates Fog computing and IoT devices.	x	x
System uses Dijkstra's algorithm to find the safest route.	x	x
System outputs safe movement directions from node locations to exits.	x	x
System controls isolated nodes (or nodes without exits) and automatically reports to the control centre for priority in rescue operations.		x
System continuously updates in real time when an incident occurs.		x

We meticulously analyze the results and discuss their implications, shedding light on the system's efficacy and performance. Next, Section 5 is going to outline the ongoing research and development efforts related to iTEVAC. We discuss potential future enhancements and applications, emphasizing our commitment to continuous improvement and adaptation to evolving emergency response needs. Finally, in Section 6, we summarize the key contributions of this paper and underscore the significance of iTEVAC in advancing emergency evacuation procedures. We reiterate its potential to save lives, reduce harm, and revolutionize evacuation strategies in urban environments.

II. Related work

The development of efficient evacuation systems has been a topic of significant interest in the field of emergency response. This section reviews the related work in this area, beginning with traditional evacuation systems, moving on to IoT-based systems like TEVAC, then discussing the role of Fog computing, followed using smartphones as well as hybrid approach and finally considering the potential for machine learning integration.

Traditional evacuation systems have relied heavily on static signage and human direction. These methods, while effective in smaller or less complex environments, often fall short in large commercial centres, older buildings, and crowded areas. The complexity of these environments, coupled with high population densities, can lead to confusion and inefficiency during emergencies. In their work (Gelenbe & Wu, 2012), the authors discuss the evolution of emergency evacuation systems and the need for heterogenous cyber-physical emergency evacuation systems. Another study (Ibrahim et al., 2016) provides a brief review on the evacuation navigation path and the need for intelligent evacuation systems. The limitations of these traditional systems underscore the need for more advanced, dynamic, and intelligent evacuation strategies.

Secondly, the Trusted Evacuation System (TEVAC) (Vinh et al., 2023) represented a significant advancement in this field. TEVAC leveraged Fog computing and IoT devices to provide real-time guidance during emergencies. However, while TEVAC has proven effective, it also highlighted challenges such as the need for faster decision-making, better handling of isolated nodes, and more comprehensive strategies for firefighting and rescue intervention. Other IoT-based systems have similarly sought to address these challenges, but none have fully succeeded in optimizing emergency evacuation procedures. For instance, Park et al. discusses an IoT-based fire evacuation system (Park et al., 2018). Other study (Kodali & Yerroju, 2017) discuss the design of a system to improve evacuation. A recent study titled 'A Fog-Based Emergency Evacuation System for Indoor Environments Using Visible Light Sensors' (Nikoloudakis et al., 2016) discusses a fog-based emergency evacuation system that uses visible light sensors for indoor

environments. Another work (Ortiz-Garcés et al., 2023) presents a prototype of an emergency response system that leverages the Internet of Things (IoT) in a fog computing environment. The system is designed to respond to emergencies in urban areas, particularly those involving vehicles.

In recent years, researchers have explored IoT-based smart evacuation systems in depth. Various studies have proposed systems that can track employee movements, detect flames, control temperature, detect smoke or gas, direct people to safety, and inform emergency services of missing employees' whereabouts (Zuolkernan et al., 2019). For instance, one study proposed a smart evacuation alert system that detects flames, controls temperature, and detects smoke or gas, overcoming the limitations of traditional evacuation systems (Nguyen et al., 2022). Khan et al. presented an intelligent fire evacuation system that utilizes IoT technologies to guide people to safety in case of a fire accident in modern, complex buildings (Khan et al., 2018). Furthermore, Gaur et al. presented a fire prevention system that uses IoT techniques and artificial intelligence to identify and prevent electrical fires (Gaur et al., 2022). Cheung et al. developed a real-time construction safety monitoring system that integrates wireless sensor network (WSN) and building information modeling (BIM) technologies to monitor hazardous gas in construction sites (Cheung et al., 2018). In another study, the authors described a conceptual design of an IoT-aided building fire evacuation control system that detects and monitors fire emergencies, provides guidance to occupants, and supports firefighting operations (Fang et al., 2021). Lastly, Sasil and Sood proposed a panic-oriented smart evacuation approach for post-disaster management using a Fog-Cloud centric IoT-based system, which includes a real-time based approach for remotely identifying the Panic Health Status (PHS) of individuals through the cyber subsystem of the system at the Fog layer (Sahil & Sood, 2022).

Next, the paper (Mohammadiounotikandi et al., 2023) proposes a hybrid nature-inspired optimization algorithm (EPC-PSO) for smart buildings' fire evacuation and control systems based on the Internet of Things (IoT). The proposed method leverages real-time fire-ground data, such as the nature of the fire and the people, to make escape decisions. The method also uses a particle idea of PSO to update the penguins' positions. Experimental results showed that the proposed method was executed accurately and effectively by cost, energy consumption, and execution time-related challenges to ensure minimum life and resource casualties. The method has decreased the execution time and cost by 10.41% and 25% compared to other algorithms. In addition, Al-Nabhan et al. (2019) present a hybrid approach for dynamic emergency evacuation using IoT and combines it with cloud computing. The proposed approach attempts to utilize the advantages of both infrastructures (IoT and cloud) to calculate evacuation paths in real time during emergency evacuation. It attempts to maximize the safety of the suggested evacuation paths by adapting to the characteristics of the hazard, evacuees' behaviour, and environmental conditions. The approach covers two emergency navigators: A Localized Emergency Navigator (LEN) and a High-Risk Emergency Navigator (HREN). Depending on some defined emergency factors, the approach decides which navigator to execute. It also handles diversified evacuation issues, such as when people get locked in a safe, however, a dead-end area of a building under emergency. Performance evaluation shows that the proposed approach achieves better outcomes in terms of survival rate and evacuation efficiency compared to the traditional algorithm.

Finally, the role of smartphones in emergency evacuation has also been explored. Smartphones, with their ubiquity and advanced capabilities, offer a promising platform for enhancing evacuation systems. For instance, Johnston et al. discusses the use of smartphones for communication among surgical teams during emergencies (Johnston et al., 2015). In the machine learning context, the potential for machine learning integration in emergency evacuation systems is a promising avenue for future research. Machine learning algorithms can analyze complex data and make predictions, which can be used to guide evacuation strategies. For instance, the authors in (Levin et al., 2018; Raita et al., 2019) discuss the use of machine learning for triage in emergency departments. This study highlights the potential for machine learning integration in emergency evacuation systems, a feature that iTEVAC seeks to incorporate in future research.

III. iTEVAC: isolation and fight

To fully appreciate the advancements introduced by iTEVAC, it is essential to revisit its predecessor, TEVAC: Trusted Evacuation System based on Fog Computing. Both systems share the same foundational context, problem definitions, and assumptions, but they diverge significantly in their outcomes and performance metrics.

A. Contextual framework

The research landscape for this study is an indoor evacuation scenario that comprises a network of 20 path-nodes intersecting with 5 strategically placed exit-nodes, referred to as 'fogs.' These fogs are technologically equipped with intelligent navigation systems, a variety of sensors, cameras, and other IoT devices capable of real-time situational analysis. The entire network is centrally orchestrated by a master node, as previously described in (Le & Huan, 2020). In the original TEVAC model (Figure 2), these intelligent nodes are not merely passive elements; they are equipped with smart navigation guides, sensors, and cameras that actively evaluate emergency conditions. They can

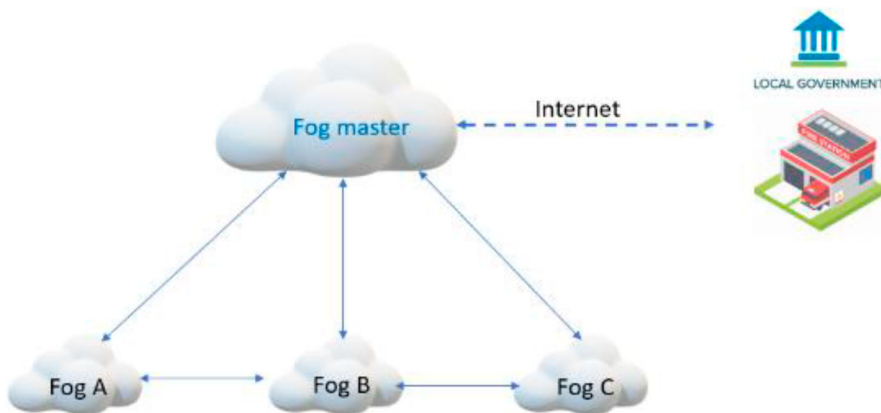


Figure 2. TEVAC based Fog computing.

instantly relay critical information to building occupants and, if necessary, connect with local emergency services for immediate intervention. Internet connectivity is maintained through the building’s Wi-Fi infrastructure, allowing users to access additional online resources during an emergency.

B. Problem statement and assumptions

The indoor escape scenario is mathematically modelled as a graph, featuring 20 intersecting nodes labelled 1–20 and 5 exit gates labelled referred to as ‘fogs’ in Figure 3. In Figure 4, we assume that nodes 1-3-5-7-10 are the exits.

In the TEVAC system, intelligent nodes equipped with navigation guides, sensors, and cameras can evaluate critical conditions and instantly provide relevant information to users. In case of an emergency, the system can connect with local authorities for prompt assistance, and users can access the internet or other services through the building’s Wi-Fi. With the intelligent instructions and optimal escape routes available at each node, individuals can evacuate safely and effectively during a crisis. In Figure 5 represents the architecture of TEVAC.

Typically, users can rely on their smartphones or the smart navigation at specific nodes for guidance in case of any trouble. If someone does not have a smartphone, they can still follow the smart instructions to find the optimal escape route. A Trusted Path $TP_t(i, j)$ is calculated at any given time t to determine the most reliable evacuation route between any two nodes or between a node and an exit. This calculation incorporates multiple variables:

$$TP_t(i, j) = \alpha_1 D_t(i, j) + \alpha_2 T_t(i, j) + \alpha_3 C_t(i, j) + \alpha_4 O_t(i, j) \tag{1}$$

Where, $D_t(i, j)$ is the physical distance between points i and j . This factor takes into account the length of the path connecting the two points.

$T_t(i, j)$: The disaster impact at time t between points i and j . This factor accounts for the severity of the disaster in the area, such as the intensity of a flood, earthquake, or fire as well as toxic gas. In addition, we assumed a set of environmental indicators such as heat, CO₂, CO, and HCN, collectively represented $T_t(i, j)$. These indicators are critical as they can

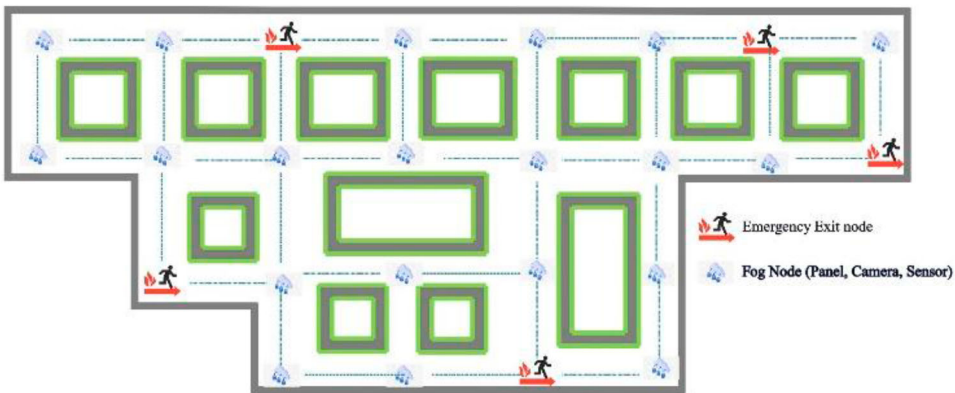


Figure 3. Indoor escape scene context.

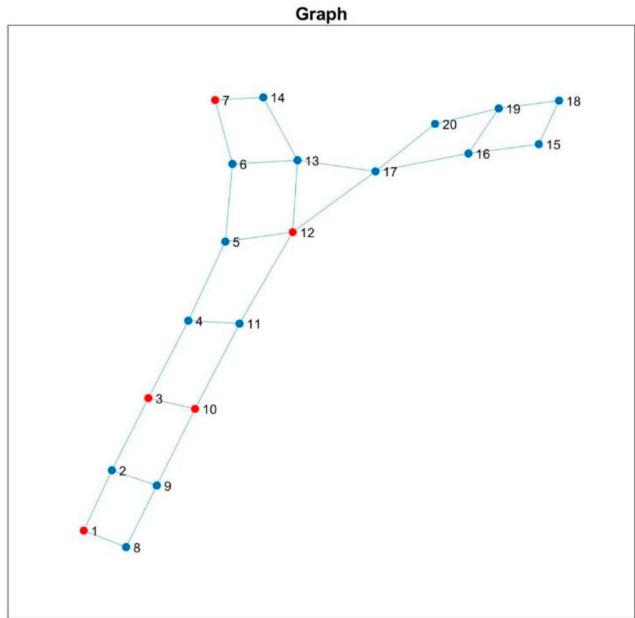


Figure 4. Graph based model.

have immediate harmful effects on human life (*Immediately Dangerous To Life or Health (IDLH) Values, n.d.; OSHA Occupational Chemical Database-Documentation of the Threshold Limit Values (TLVs) and Biological Exposure Indices (BEIs) - Carbon Dioxide, n.d.*).

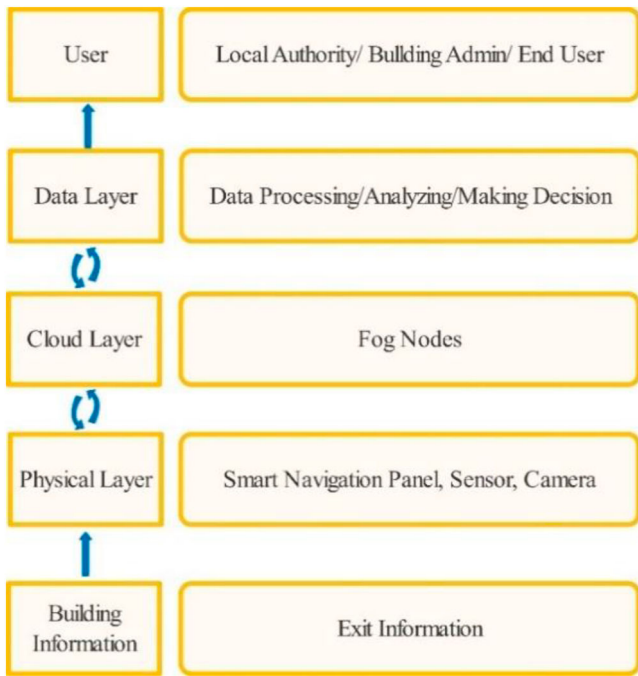


Figure 5. Overall architecture of Trusted Evacuation system.

$C_t(i, j)$: The population density at time t between points i and j . This factor considers the number of people present in the area at the time of evacuation.

$O_t(i, j)$: The obstacle density at time t between points i and j . This factor considers the number and severity of obstacles, such as debris or damaged infrastructure, along the path.

These variables are weighted by coefficients α_i , which are empirically determined and satisfy the condition $\sum \alpha_i = 1$. They can be adjusted based on the specific context of the evacuation scenario or the priorities of the emergency responders. For the scope of TEVAC as well as iTEVAC, we assumed that a Trusted Path value must be lower than 0.8 to be considered safe for navigation. The issue at hand aligns with the proposal, which involves identifying an optimal solution for safely evacuating individuals to designated exit gates in the event of emergencies within a commercial centre. However, this section will address the critical aspect of issuing emergency alerts for isolated locations where no viable escape routes are available.

Drawing inspiration from firefighting strategies, we propose an improved version of the TEVAC model, referred to as iTEVAC. iTEVAC incorporates the principles of isolation and firefighting as well as rescue intervention to enhance its performance. By integrating these elements, the iTEVAC model operates more efficiently and achieves faster processing times for improved results. The overall steps can be described as the following [Figure 6](#).

C. The proposed enhanced model

Proposal for the enhanced iTEVAC algorithm for finding safe evacuation routes in a commercial centre during emergencies, including the following steps:

1. Input

- Grid matrix of the commercial centre (M00)
- Matrix containing the nodes requiring movement within the commercial centre (Node)
- Matrix containing the entrance/exit gates of the commercial centre (Out)
- Maximum objective function value (Max)
- Threshold value for the objective function (Thres)

2. Output

- Grid matrix of the commercial centre after removing paths with objective function values exceeding the threshold (M)
- Matrix containing the isolated nodes within the commercial centre after removing paths with objective function values exceeding the threshold (E)
- Time required to perform the algorithm (time)
- Matrix containing the safest path from the nodes requiring movement in the commercial centre to the entrance/exit gates (optimal)
- Matrix containing all possible paths from the nodes requiring movement to all entrance/exit gates within the commercial centre (outpath)

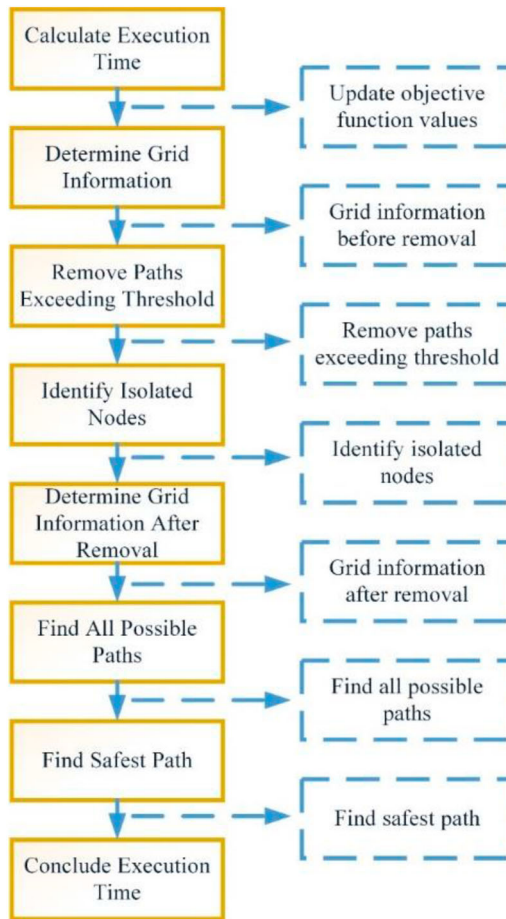


Figure 6. The iTEVAC process.

- Information about the grid graph of the commercial centre before removing paths with objective function values exceeding the threshold (G_0 : containing objective function values of paths, G_{0_new} : not containing objective function values of paths)
- Information about the grid graph of the commercial centre after removing paths with objective function values exceeding the threshold (G : containing objective function values of paths, G_new : not containing objective function values of paths)

3. Processing steps

- **Step 1:** Begin by calculating the execution time of the algorithm and updating the objective function values from the paths within the commercial centre based on M_{00} (M_0).
 - If the value of $M_{00}(i,j)$ in the matrix is non-zero and not equal to Max, assign the value of $M_0(i,j) = M_0(j,i)$ in the M_0 matrix as a randomly generated value within the range $[0-100]$.
 - Otherwise, assign the value of $M_0(i,j)$ in the M_0 matrix as the value of $M_{00}(i,j)$ in the M_{00} matrix.

- Step 2: Based on M_0 , determine the grid information of the commercial centre before removing paths with objective function values exceeding the threshold ($G_0 = I_0 - F_0 - K_0$: containing objective function values of paths, $G_{0_new} = I_0 - F_0$: not containing objective function values of paths. I_0 is the matrix containing the starting nodes of paths that can be executed between nodes within the commercial centre, where the starting node is always smaller than the destination node. F_0 is the matrix containing the destination nodes of paths that can be executed between nodes within the commercial centre, where the starting node is always smaller than the destination node. K_0 is the matrix containing the objective function values of paths that can be executed between nodes within the commercial centre, where the starting node is always smaller than the destination node.)
 - If the destination node j is greater than the starting node i and the value of $M_0(i,j)$ is not equal to Max, set I_0 to i , F_0 to j , and K_0 to $M_0(i,j)$.
- Step 3: Based on M_0 , remove paths with objective function values exceeding the threshold (M).
 - If $M_0(i,j)$ is non-zero, not equal to Max, and greater than threshold, set $M(i,j)$ to Max.
 - Otherwise, set $M(i,j)$ to $M_0(i,j)$.
- Step 4: Based on M , identify the isolated nodes within the commercial centre after removing paths with objective function values exceeding the threshold (E).
 - If all elements in the row $M(i,:)$ are either zero or equal to Max, and i is different from the elements in the Out matrix, issue an alert indicating that node i is isolated (E).
- Step 5: Based on M , determine the grid information of the commercial centre after removing paths with objective function values exceeding the threshold ($G = I - F - K$: containing objective function values of paths, $G_{_new} = I - F$: not containing objective function values of paths. I is the matrix containing the starting nodes of paths that can be executed between nodes within the commercial centre, where the starting node is always smaller than the destination node. F is the matrix containing the destination nodes of paths that can be executed between nodes within the commercial centre, where the starting node is always smaller than the destination node. K is the matrix containing the objective function values of paths that can be executed between nodes within the commercial centre, where the starting node is always smaller than the destination node.)
 - If the destination node j is greater than the starting node i and the value of the elements $M(i,j)$ is not equal to Max, set I to i , F to j , and K to $M(i,j)$.
- Step 6: Based on G , find all possible paths from the nodes requiring movement to all entrance/exit gates within the commercial centre (outpath).
 - Use the Dijkstra's algorithm to find the most optimal path between any two nodes in the grid of the commercial centre, with the requirement that the sum of the objective function values of the paths within the entire path is minimized (outpath).
- Step 7: Based on M and outpath, find the safest path from the nodes requiring movement within the commercial centre to the entrance/exit gates (optimal).
 - Calculate the objective function value of the optimal path between any two obtained nodes in outpath.
 - Find the minimum of the objective function values for each moving node in Note with the gates in Out to determine the optimal escape route for each moving node (optimal).

– **Step 8:** Conclude the execution time of the algorithm (time).

The iTEVAC algorithm provides an efficient and effective approach to finding safe evacuation routes in a commercial centre during emergencies, considering the objective function values and threshold constraints. Its summary is described as [Table 2](#) below.

D. Assumption and experimental work

Before delving into the proposed enhanced iTEVAC algorithm, which aims to identify secure evacuation routes during emergency situations in a commercial centre, it is essential to provide an overview of the problem and the underlying assumptions of this research. As discussed earlier, we assume that the commercial centre is equipped with a grid-based infrastructure consisting of interconnected nodes that can be utilized for navigation and evacuation purposes. Moreover, our approach considers the objective function values associated with various paths within the commercial centre, with the goal of optimizing evacuation routes while considering a predefined threshold value. Additionally, we assume that the architectural layout of the commercial centre is represented by the illustrated diagram in [Figure 3](#).

To implement the proposed algorithm, we select the parameter values as presented in [Table 3](#).

We then proceed with the processing steps at four different time intervals using the proposed algorithm, as outlined in [Table 4](#).

To facilitate further analysis and computations, **Step 01** involves transforming the architectural diagram of the commercial centre depicted in [Figure 3](#) into a grid matrix representation, denoted as M_0 , as shown in [Table 5](#).

In the grid matrix, the symbol ' ' represents a path that is not formed in the grid network, '0' represents the node itself ('1' \rightarrow '1', ...), and 'x' represents the updated value from the sensor. At time t_1 , the intelligent system receives sensor values as depicted in [Figure 7](#).

To ensure the safe evacuation of individuals to designated exit gates and issue emergency alerts for isolated locations without escape routes, the control programme proceeds to Step 02.

Step 02: Update the grid matrix ([Table 6](#)) In this step, the grid matrix is updated based on certain criteria. The specific details of the update process are depicted in [Table 6](#).

This step involves modifying the grid matrix by incorporating new information or applying specific rules. The resulting grid matrix, shown in [Table 6](#), reflects the necessary updates to facilitate the subsequent stages of the algorithm.

Moving forward, the algorithm continues with the next steps to further analyze and process the grid matrix for identifying secure evacuation routes and addressing isolated locations without escape routes.

In Step 3, the programme checks for specific connections that exceed a predefined threshold by comparing the sensor values. If a sensor value is greater than 80, the corresponding path is removed from the grid matrix, as illustrated in [Table 7](#). This step is carried out using the following algorithm 1.

Table 2. Overall iTEVAC algorithm.

Step	Description
1	<p>Calculate the execution time (time). Update the objective function values from the paths within the commercial centre based on M00: – For each element M00(i,j) in the grid matrix: – If M00(i,j) ≠ 0 and M00(i,j) ≠ Max: – Set M0(i,j) = M0(j,i) to a randomly generated value in the range [0, 100]. – Else: – Set M0(i,j) = M00(i,j).</p>
2	<p>Determine the grid information of the commercial centre before removing paths with objective function values exceeding the threshold: – Initialize G0 = I0 - F0 - K0 and G0_new = I0 - F0. – For each element M0(i,j) in the grid matrix: – If M0(i,j) ≠ 0 and M0(i,j) ≠ Max: – Set I0 = i, F0 = j, and K0 = M0(i,j).</p>
3	<p>Remove paths with objective function values exceeding the threshold: – Initialize M as an empty grid matrix. – For each element M0(i,j) in the grid matrix: – If M0(i,j) ≠ 0 and M0(i,j) ≠ Max and M0(i,j) > Thres: – Set M(i,j) = Max. – Else: – Set M(i,j) = M0(i,j).</p>
4	<p>Identify the isolated nodes within the commercial centre after removing paths with objective function values exceeding the threshold: – Initialize E as an empty matrix. – For each node i in the grid matrix: – If all elements in row M(i,:) are 0 or equal to Max and i ∉ Out: – Mark node i as isolated in matrix E.</p>
5	<p>Determine the grid information of the commercial centre after removing paths with objective function values exceeding the threshold: – Initialize G = I - F - K and G_new = I - F. – For each element M(i,j) in the grid matrix: – If M(i,j) ≠ 0 and M(i,j) ≠ Max: – Set I = i, F = j, and K = M(i,j).</p>
6	<p>Find all possible paths from the nodes requiring movement to all entrance/exit gates within the commercial centre: – Calculate the optimal paths using Dijkstra's algorithm and store them in the matrix outpath.</p>
7	<p>Find the safest path from the nodes requiring movement within the commercial centre to the entrance/exit gates: – Calculate the objective function value for each path in outpath. – For each node in Node: – Find the minimum objective function value among the paths in outpath connected to the corresponding gate in Out and store it in matrix optimal.</p>
8	<p>Conclude the execution time of the algorithm (time).</p>

Algorithm 1 Remove Hazardous Paths

Inputs: Matrix M0 (Original grid matrix); Max (Maximum value); N (Size of the matrix)

Outputs: Matrix M (Updated grid matrix with hazardous paths removed)

Procedure:

1. Initialize M as M0.
 2. For each element in M0:
 - a. If M0(i,j) > 80, set M(i,j) to Max.
 - b. Otherwise, keep the value of M0(i,j) in M(i,j).
 3. Return the updated matrix M.
-

Table 3. Parameter values of the proposed algorithm.

Parameter	Value
x	100*rand(1)
Max	2*10 ⁶
Note	[2 4 5 6 8 9 11 13 14 15 16 17 18 19 20]
Out	[1 3 7 10 12]
Threshold	80

Table 4. Scenario processing steps.

Step	Input	Output
1	M00, Note, Out, Max, Threshold	M1, E1, time1, optimal1, outpath1, G01, G1, G01_new, G1_new
2	M1, Note, Out, Max, Threshold	M2, E2, time2, optimal2, outpath2, G02, G2, G02_new, G2_new
3	M2, Note, Out, Max, Threshold	M3, E3, time3, optimal3, outpath3, G03, G3, G03_new, G3_new
4	M3, Note, Out, Max, Threshold	M4, E4, time4, optimal4, outpath4, G04, G4, G04_new, G4_new

Afterwards, in Step 4, assuming that the emergency exits are located at the nodes specified in the Out array [1 3 7 10 12] and the remaining nodes are listed in the Note array [2 4 5 6 8 9 11 13 14 15 16 17 18 19 20], we proceed to identify the isolated node(s) for issuing emergency alerts using Algorithm 2 as follows.

Algorithm 2 Identify Isolated Nodes

Inputs: Matrix M (Updated grid matrix with hazardous paths removed); Note (List of nodes requiring movement within the commercial centre); Out (List of entrance/exit gates of the commercial centre)

Outputs: Matrix E (List of isolated nodes within the commercial centre)

Procedure:

1. Initialize an empty array E to store the isolated nodes.
 2. For each node i in the list Note:
 - a. Initialize an array T with all elements set to 0.
 - b. For each node j in the range from 1 to N:
 - If $M(i, j)$ is equal to 0 or $M(i, j)$ is equal to the maximum value Max, set $T(j)$ to 0.
 - Otherwise, set $T(j)$ to 1.
 - c. Calculate the sum of all elements in array T and store it in $R(i)$.
 3. Initialize a variable r as 0 to keep track of the number of isolated nodes.
 4. For each node i in the range from 1 to N:
 - If $R(i)$ is equal to 0, increment r by 1 and store the value of i in the r -th position of array E.
-

After that, in Step 5, the grid matrix M is converted into sub-matrices I, F, and K to utilize the Dijkstra's algorithm for finding the shortest path between any two nodes in the network. It is important to note that due to node 8 being isolated, the grid matrix is divided into two separate grids as in Figure 8. Algorithm 3 is used for this conversion.

Algorithm 3 Convert Grid Matrix to Sub-Matrices

Inputs: Matrix M (Updated grid matrix with hazardous paths removed); N (Size of the matrix)

Outputs: Matrix I (Sub-matrix containing starting nodes for possible paths); Matrix F (Sub-matrix containing destination nodes for possible paths); Matrix K (Sub-matrix containing objective function values for possible paths)

Procedure:

1. Initialize a counter variable r to 0.
 2. For each node i in the range from 1 to the total number of nodes:
 - For each node j in the range from 1 to the total number of nodes:
 - If j is greater than 1 and $M(i, j)$ is not equal to Max:
 - Increment the counter variable r by 1.
-

(Continued)

Table 5. Grid matrix representation of the commercial center’ nodes.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1	0	x	Max	Max	Max	Max	Max	x	Max	Max	Max	Max	Max	Max	Max	Max	Max	Max	Max	Max
2	x	0	x	Max	Max	Max	Max	Max	x	Max	Max	Max	Max	Max	Max	Max	Max	Max	Max	Max
3	Max	x	0	x	Max	Max	Max	Max	x	Max	Max	Max	Max	Max	Max	Max	Max	Max	Max	Max
4	Max	Max	x	0	x	Max	Max	Max	Max	Max	x	Max	Max	Max	Max	Max	Max	Max	Max	Max
5	Max	Max	Max	x	0	x	Max	Max	Max	Max	Max	x	Max	Max	Max	Max	Max	Max	Max	Max
6	Max	Max	Max	Max	x	0	x	Max	Max	Max	Max	Max	x	Max	Max	Max	Max	Max	Max	Max
7	Max	Max	Max	Max	Max	x	0	Max	Max	Max	Max	Max	x	Max	Max	Max	Max	Max	Max	Max
8	x	Max	Max	Max	Max	Max	Max	0	x	Max	Max	Max	Max	Max	Max	Max	Max	Max	Max	Max
9	Max	x	Max	Max	Max	Max	Max	x	0	x	Max	Max	Max	Max	Max	Max	Max	Max	Max	Max
10	Max	Max	x	Max	Max	Max	Max	Max	x	0	x	Max	Max	Max	Max	Max	Max	Max	Max	Max
11	Max	Max	Max	x	Max	Max	Max	Max	Max	x	0	x	Max	Max	Max	Max	Max	Max	Max	Max
12	Max	Max	Max	Max	x	Max	Max	Max	Max	Max	x	0	x	Max	Max	Max	x	Max	Max	Max
13	Max	Max	Max	Max	Max	x	Max	Max	Max	Max	Max	x	0	x	Max	Max	x	Max	Max	Max
14	Max	Max	Max	Max	Max	Max	x	Max	Max	Max	Max	Max	x	0	Max	Max	Max	Max	Max	Max
15	Max	Max	Max	Max	Max	Max	Max	Max	Max	Max	Max	Max	Max	0	x	Max	x	Max	Max	Max
16	Max	Max	Max	Max	Max	Max	Max	Max	Max	Max	Max	Max	Max	Max	x	0	x	Max	x	Max
17	Max	Max	Max	Max	Max	Max	Max	Max	Max	Max	Max	x	x	Max	Max	x	0	Max	Max	x
18	Max	Max	Max	Max	Max	Max	Max	Max	Max	Max	Max	Max	Max	Max	x	Max	Max	0	x	Max
19	Max	Max	Max	Max	Max	Max	Max	Max	Max	Max	Max	Max	Max	Max	Max	x	Max	x	0	x
20	Max	Max	Max	Max	Max	Max	Max	Max	Max	Max	Max	Max	Max	Max	Max	Max	x	Max	x	0

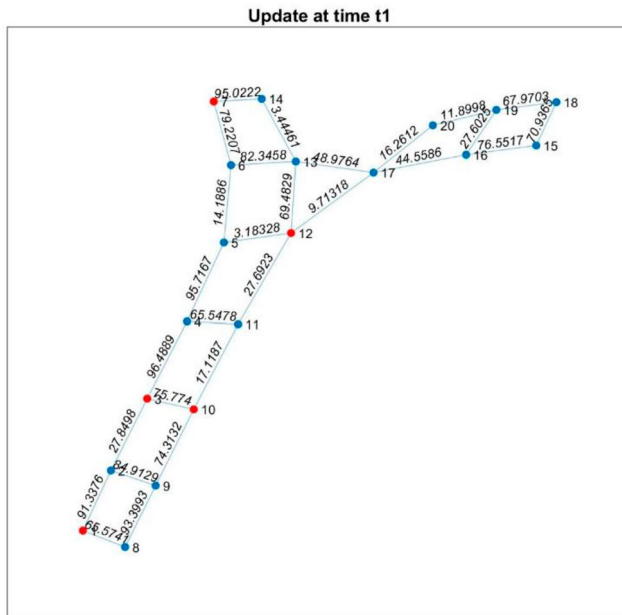


Figure 7. Sensor values received by the intelligent system at time t1.

Continued.

Algorithm 3 Convert Grid Matrix to Sub-Matrices

- Store the value of i in the corresponding position of matrix I .
- Store the value of j in the corresponding position of matrix F .
- Store the value of $M(i,j)$ in the corresponding position of matrix K .

3. Return the sub-matrices I , F , and K .

By executing Algorithm 3, the grid matrix M is transformed into sub-matrices I , F , and K , enabling the application of the Dijkstra's algorithm to find the shortest paths between any two nodes in the network. It is important to note that due to the isolation of node 8, the grid matrix is divided into separate sub-matrices, ensuring accurate path calculations within each grid.

Next, in Step 6 the programme finds the optimal path between any two nodes in the network using the 'shortestpath' function and calculates the objective function values for the obtained optimal paths. This is achieved through Algorithm 4 and Algorithm 5, resulting in Table 8(A–D) and Table 9, respectively.

Algorithm 4 Finding the Optimal Paths

Inputs: Matrix I (Sub-matrix containing starting nodes for possible paths); Matrix F (Sub-matrix containing destination nodes for possible paths); Matrix K (Sub-matrix containing objective function values for possible paths)

Outputs: Cell array outpath: Contains the optimal paths between each pair of nodes

Procedure:

1. Create a graph G using the sub-matrices I , F , and K .
 2. For each node i in the range from 1 to N :
 - For each node j in the range from 1 to N :
 - Find the shortest path between node i and node j using the 'shortestpath' function.
 - Store the obtained path in the cell array outpath at the position (i, j) .
 3. Return the cell array outpath.
-

Table 6. Updated grid matrix reflecting changes in the network.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1	0	91.33	2e+6	2e+6	2e+6	2e+6	2e+6	65.57	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6
2	91.33	0	27.84	2e+6	2e+6	2e+6	2e+6	2e+6	84.91	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6
3	2e+6	27.84	0	96.48	2e+6	2e+6	2e+6	2e+6	2e+6	75.77	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6
4	2e+6	2e+6	96.48	0	95.71	2e+6	2e+6	2e+6	2e+6	2e+6	65.54	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6
5	2e+6	2e+6	2e+6	95.71	0	14.18	2e+6	2e+6	2e+6	2e+6	2e+6	03.18	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6
6	2e+6	2e+6	2e+6	2e+6	14.18	0	79.22	2e+6	2e+6	2e+6	2e+6	2e+6	82.34	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6
7	2e+6	2e+6	2e+6	2e+6	2e+6	79.22	0	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	95.02	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6
8	65.57	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	0	93.39	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6
9	2e+6	84.91	2e+6	2e+6	2e+6	2e+6	2e+6	93.39	0	74.31	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6
10	2e+6	2e+6	75.77	2e+6	2e+6	2e+6	2e+6	2e+6	74.31	0	17.11	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6
11	2e+6	2e+6	2e+6	65.54	2e+6	2e+6	2e+6	2e+6	2e+6	17.11	0	27.69	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6
12	2e+6	2e+6	2e+6	2e+6	03.18	2e+6	2e+6	2e+6	2e+6	2e+6	27.69	0	69.48	2e+6	2e+6	2e+6	09.71	2e+6	2e+6	2e+6
13	2e+6	2e+6	2e+6	2e+6	2e+6	82.34	2e+6	2e+6	2e+6	2e+6	2e+6	69.48	0	03.44	2e+6	2e+6	48.97	2e+6	2e+6	2e+6
14	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	95.02	2e+6	2e+6	2e+6	2e+6	2e+6	03.44	0	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6
15	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	0	76.55	2e+6	70.93	2e+6	2e+6
16	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	76.55	0	44.55	2e+6	27.60	2e+6
17	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	09.71	48.97	2e+6	2e+6	44.55	0	2e+6	2e+6	16.26
18	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	70.93	2e+6	2e+6	0	67.97	2e+6
19	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	27.60	2e+6	67.97	0	11.89
20	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	16.26	2e+6	11.89	0

Table 7. Removing the hazardous paths in the grid matrix by replacing the values exceeding the threshold with the maximum value.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1	0	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	65.57	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6
2	2e+6	0	27.84	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6
3	2e+6	27.84	0	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	75.77	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6
4	2e+6	2e+6	2e+6	0	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	65.54	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6
5	2e+6	2e+6	2e+6	2e+6	0	14.18	2e+6	2e+6	2e+6	2e+6	2e+6	03.18	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6
6	2e+6	2e+6	2e+6	2e+6	14.18	0	79.22	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6
7	2e+6	2e+6	2e+6	2e+6	2e+6	79.22	0	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6
8	65.57	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	0	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6
9	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	0	74.31	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6
10	2e+6	2e+6	75.77	2e+6	2e+6	2e+6	2e+6	2e+6	74.31	0	17.11	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6
11	2e+6	2e+6	2e+6	65.54	2e+6	2e+6	2e+6	2e+6	2e+6	17.11	0	27.69	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6
12	2e+6	2e+6	2e+6	2e+6	03.18	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	27.69	0	69.48	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6
13	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	69.48	0	03.44	2e+6	2e+6	48.97	2e+6	2e+6	2e+6
14	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	03.44	0	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6
15	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	0	76.55	2e+6	70.93	2e+6	2e+6
16	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	76.55	0	44.55	2e+6	27.60	2e+6
17	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	48.97	2e+6	2e+6	44.55	0	2e+6	16.26
18	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	70.93	2e+6	2e+6	0	67.97
19	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	67.97	0
20	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	16.26	2e+6	11.89	0

Algorithm 5 Calculating Objective Function Values for Optimal Paths

Inputs: Matrix M (Updated grid matrix with hazardous paths removed); Cell array outpath (Contains the optimal paths between each pair of nodes)

Outputs: Matrix Fitness (Contains the objective function values for the optimal paths)

Procedure:

1. Initialize the matrix Fitness with zeros.
2. For each node i in the range from 1 to N:
 - For each node j in the range from 1 to N:
 - Retrieve the optimal path H from the cell array outpath at position (i, j).
 - Get the size of H and assign it to variables x and y.
 - Initialize a vector HH with zeros of size (1, y-1).
 - For each index k in the range from 1 to y-1:
 - Assign the value $M(H(k), H(k + 1))$ to the k-th element of HH.
 - Calculate the sum of the elements in HH and assign it to the (i, j)-th element of Fitness.
3. Return the matrix Fitness.

By executing Algorithm 4 and Algorithm 5, we obtain the optimal paths between each pair of nodes in the network and calculate the corresponding objective function values.

In Step 7, the algorithm computes the optimal evacuation paths for the nodes specified in Note = [2, 4, 5, 6, 8, 9, 11, 13, 14, 15, 16, 17, 18, 19, 20] towards the designated exit gates Out = [1, 3, 7, 10, 12], employing Algorithm 6 as delineated below.

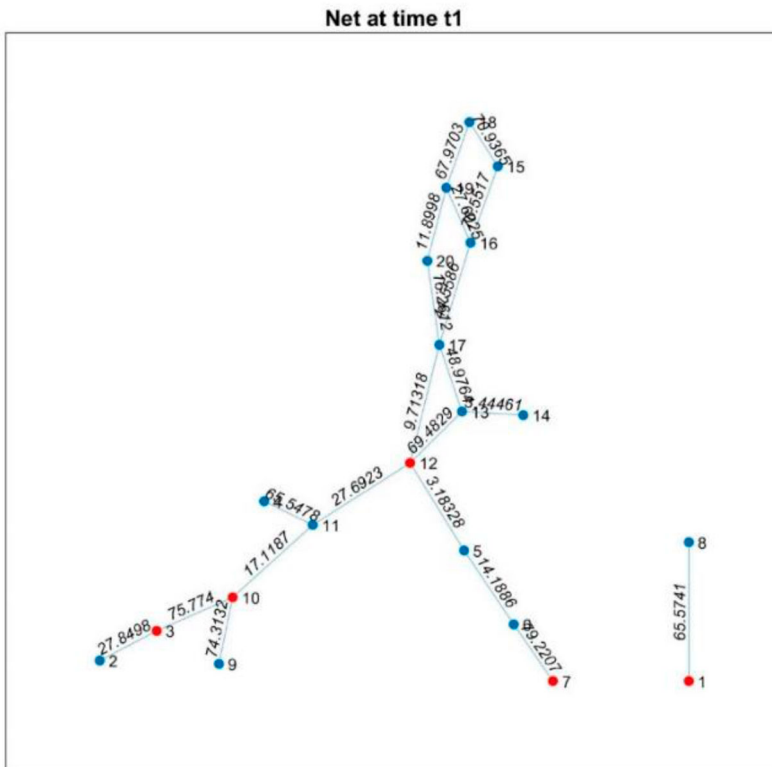


Figure 8. Separated grids.

Algorithm 6 Compute Optimal Evacuation Paths to Exit Gates

Inputs: Fitness (Matrix containing objective function values for optimal paths); Outpath (Matrix storing optimal paths between nodes); Note (Array of nodes requiring evacuation); Out (Array of designated exit gates); N (Dimension of the matrix)

Output: Optimal (A matrix containing the optimal evacuation paths for each node in Note to the nearest exit in Out)

Procedure:

Initialize FF as a matrix populated with large values (e.g. 10^{10}) to represent infinity.

Ascertain the dimensions of Note and assign them to variables x and y.

Ascertain the dimensions of Out and assign them to variables m and n.

For each node i in Note and each exit gate j in Out:

Assign $Fitness(Note(i), Out(j))$ to $FF(Note(i), Out(j))$.

For each node i in Note:

Identify the minimum value in the row $FF(Note(i), :)$ and assign it to variables a and b.

Assign $outpath\{Note(i), b\}$ to $optimal\{i\}$.

Output the matrix optimal containing the optimal paths.

The computed optimal paths for each node are tabulated in [Table 10](#).

Lastly, in Step 8, should the sensor values undergo an update at time t_2 , the procedure will recommence from Step 3 through Step 7. Within iTEVAC, the decomposition of the grid into subgrids enables individuals in each subgrid to swiftly pinpoint the nearest secure routes, thereby circumventing nodes that become isolated due to emergency situations. This is anticipated to expedite the evacuation process while bolstering safety. Moreover, this cyclical process guarantees that the evacuation routes are dynamically updated in response to real-time sensor data, thereby ensuring the most effective safety protocols are in place for occupants within the commercial centre.

IV. Result and discussion

In general, the iTEVAC model operates through a systematic process that can be categorized into three primary stages: input, output, and processing steps. In the input stage, the model ingests a variety of matrices that serve as representations of the commercial centre. These matrices encompass the nodes that require movement, the entrance and exit gates, and the objective function values. The output stage of the model generates several key results. These include an updated grid matrix, a matrix of isolated nodes, the time required for the algorithm's execution, a matrix delineating the safest paths, and information about the grid graph of the commercial centre both prior to and following the removal of paths with objective function values that exceed the threshold.

The processing steps form the core of the iTEVAC model's operations. The model initiates by calculating the execution time of the algorithm. Subsequently, it updates the objective function values and determines the grid information. The model then proceeds to remove paths with objective function values that surpass the threshold and identifies isolated nodes. Following this, the model finds all feasible paths from the nodes that require movement to all entrance and exit gates within the commercial centre. The model then identifies the safest path from the nodes requiring movement within the commercial centre to the entrance or exit gates. The final step in the process involves concluding the execution time of the algorithm. This systematic approach ensures the model's efficiency and effectiveness in identifying safe evacuation routes during emergencies.

Table 8. The value of optimal paths.

	1	2	3	4	5
1	1	∅	∅	∅	∅
2	∅	2	[2,3]	[2,3,10,11,4]	[2,3,10,11,12,5]
3	∅	[3,2]	3	[3,10,11,4]	[3,10,11,12,5]
4	∅	[4,11,10,3,2]	[4,11,10,3]	4	[4,11,12,5]
5	∅	[5,12,11,10,3,2]	[5,12,11,10,3]	[5,12,11,4]	5
6	∅	[6,5,12,11,10,3,2]	[6,5,12,11,10,3]	[6,5,12,11,4]	[6,5]
7	∅	[7,6,5,12,11,10,3,2]	[7,6,5,12,11,10,3]	[7,6,5,12,11,4]	[7,6,5]
8	[8,1]	∅	∅	∅	∅
9	∅	[9,10,3,2]	[9,10,3]	[9,10,11,4]	[9,10,11,12,5]
10	∅	[10,3,2]	[10,3]	[10,11,4]	[10,11,12,5]
11	∅	[11,10,3,2]	[11,10,3]	[11,4]	[11,12,5]
12	∅	[12,11,10,3,2]	[12,11,10,3]	[12,11,4]	[12,5]
13	∅	[13,17,12,11,10,3,2]	[13,17,12,11,10,3]	[13,17,12,11,4]	[13,17,12,5]
14	∅	[14,13,17,12,11,10,3,2]	[14,13,17,12,11,10,3]	[14,13,17,12,11,4]	[14,13,17,12,5]
15	∅	[15,16,17,12,11,10,3,2]	[15,16,17,12,11,10,3]	[15,16,17,12,11,4]	[15,16,17,12,5]
16	∅	[16,17,12,11,10,3,2]	[16,17,12,11,10,3]	[16,17,12,11,4]	[16,17,12,5]
17	∅	[17,12,11,10,3,2]	[17,12,11,10,3]	[17,12,11,4]	[17,12,5]
18	∅	[18,19,20,17,12,11,10,3,2]	[18,19,20,17,12,11,10,3]	[18,19,20,17,12,11,4]	[18,19,20,17,12,5]
19	∅	[19,20,17,12,11,10,3,2]	[19,20,17,12,11,10,3]	[19,20,17,12,11,4]	[19,20,17,12,5]
20	∅	[20,17,12,11,10,3,2]	[20,17,12,11,10,3]	[20,17,12,11,4]	[20,17,12,5]
	6	7	8	9	10
1	∅	∅	[1,8]	∅	∅
2	[2,3,10,11,12,5,6]	[2,3,10,11,12,5,6,7]	∅	[2,3,10,9]	[2,3,10]
3	[3,10,11,12,5,6]	[3,10,11,12,5,6,7]	∅	[3,10,9]	[3,10]
4	[4,11,12,5,6]	[4,11,12,5,6,7]	∅	[4,11,10,9]	[4,11,10]
5	[5,6]	[5,6,7]	∅	[5,12,11,10,9]	[5,12,11,10]
6	6	[6,7]	∅	[6,5,12,11,10,9]	[6,5,12,11,10]
7	[7,6]	7	∅	[7,6,5,12,11,10,9]	[7,6,5,12,11,10]
8	∅	∅	8	∅	∅
9	[9,10,11,12,5,6]	[9,10,11,12,5,6,7]	∅	9	[9,10]
10	[10,11,12,5,6]	[10,11,12,5,6,7]	∅	[10,9]	10
11	[11,12,5,6]	[11,12,5,6,7]	∅	[11,10,9]	[11,10]
12	[12,5,6]	[12,5,6,7]	∅	[12,11,10,9]	[12,11,10]
13	[13,17,12,5,6]	[13,17,12,5,6,7]	∅	[13,17,12,11,10,9]	[13,17,12,11,10]
14	[14,13,17,12,5,6]	[14,13,17,12,5,6,7]	∅	[14,13,17,12,11,10,9]	[14,13,17,12,11,10]

(Continued)

Table 8. Continued.

	6	7	8	9	10
15	[15,16,17,12,5,6]	[15,16,17,12,5,6,7]	∅	[15,16,17,12,11,10,9]	[15,16,17,12,11,10]
16	[16,17,12,5,6]	[16,17,12,5,6,7]	∅	[16,17,12,11,10,9]	[16,17,12,11,10]
17	[17,12,5,6]	[17,12,5,6,7]	∅	[17,12,11,10,9]	[17,12,11,10]
18	[18,19,20,17,12,5,6]	[18,19,20,17,12,5,6,7]	∅	[18,19,20,17,12,11,10,9]	[18,19,20,17,12,11,10]
19	[19,20,17,12,5,6]	[19,20,17,12,5,6,7]	∅	[19,20,17,12,11,10,9]	[19,20,17,12,11,10]
20	[20,17,12,5,6]	[20,17,12,5,6,7]	∅	[20,17,12,11,10,9]	[20,17,12,11,10]
	11	12	13	14	15
1	∅	∅	∅	∅	∅
2	[2,3,10,11]	[2,3,10,11,12]	[2,3,10,11,12,17,13]	[2,3,10,11,12,17,13,14]	[2,3,10,11,12,17,16,15]
3	[3,10,11]	[3,10,11,12]	[3,10,11,12,17,13]	[3,10,11,12,17,13,14]	[3,10,11,12,17,16,15]
4	[4,11]	[4,11,12]	[4,11,12,17,13]	[4,11,12,17,13,14]	[4,11,12,17,16,15]
5	[5,12,11]	[5,12]	[5,12,17,13]	[5,12,17,13,14]	[5,12,17,16,15]
6	[6,5,12,11]	[6,5,12]	[6,5,12,17,13]	[6,5,12,17,13,14]	[6,5,12,17,16,15]
7	[7,6,5,12,11]	[7,6,5,12]	[7,6,5,12,17,13]	[7,6,5,12,17,13,14]	[7,6,5,12,17,16,15]
8	∅	∅	∅	∅	∅
9	[9,10,11]	[9,10,11,12]	[9,10,11,12,17,13]	[9,10,11,12,17,13,14]	[9,10,11,12,17,16,15]
10	[10,11]	[10,11,12]	[10,11,12,17,13]	[10,11,12,17,13,14]	[10,11,12,17,16,15]
11	11	[11,12]	[11,12,17,13]	[11,12,17,13,14]	[11,12,17,16,15]
12	[12,11]	12	[12,17,13]	[12,17,13,14]	[12,17,16,15]
13	[13,17,12,11]	[13,17,12]	13	[13,14]	[13,17,16,15]
14	[14,13,17,12,11]	[14,13,17,12]	[14,13]	14	[14,13,17,16,15]
15	[15,16,17,12,11]	[15,16,17,12]	[15,16,17,13]	[15,16,17,13,14]	15
16	[16,17,12,11]	[16,17,12]	[16,17,13]	[16,17,13,14]	[16,15]
17	[17,12,11]	[17,12]	[17,13]	[17,13,14]	[17,16,15]
18	[18,19,20,17,12,11]	[18,19,20,17,12]	[18,19,20,17,13]	[18,19,20,17,13,14]	[18,15]
19	[19,20,17,12,11]	[19,20,17,12]	[19,20,17,13]	[19,20,17,13,14]	[19,16,15]
20	[20,17,12,11]	[20,17,12]	[20,17,13]	[20,17,13,14]	[20,19,16,15]

	16	17	18	19	20
1	[]	[]	[]	[]	[]
2	[2,3,10,11,12,17,16]	[2,3,10,11,12,17]	[2,3,10,11,12,17,20,19,18]	[2,3,10,11,12,17,20,19]	[2,3,10,11,12,17,20]
3	[3,10,11,12,17,16]	[3,10,11,12,17]	[3,10,11,12,17,20,19,18]	[3,10,11,12,17,20,19]	[3,10,11,12,17,20]
4	[4,11,12,17,16]	[4,11,12,17]	[4,11,12,17,20,19,18]	[4,11,12,17,20,19]	[4,11,12,17,20]
5	[5,12,17,16]	[5,12,17]	[5,12,17,20,19,18]	[5,12,17,20,19]	[5,12,17,20]
6	[6,5,12,17,16]	[6,5,12,17]	[6,5,12,17,20,19,18]	[6,5,12,17,20,19]	[6,5,12,17,20]
7	[7,6,5,12,17,16]	[7,6,5,12,17]	[7,6,5,12,17,20,19,18]	[7,6,5,12,17,20,19]	[7,6,5,12,17,20]
8	[]	[]	[]	[]	[]
9	[9,10,11,12,17,16]	[9,10,11,12,17]	[9,10,11,12,17,20,19,18]	[9,10,11,12,17,20,19]	[9,10,11,12,17,20]
10	[10,11,12,17,16]	[10,11,12,17]	[10,11,12,17,20,19,18]	[10,11,12,17,20,19]	[10,11,12,17,20]
11	[11,12,17,16]	[11,12,17]	[11,12,17,20,19,18]	[11,12,17,20,19]	[11,12,17,20]
12	[12,17,16]	[12,17]	[12,17,20,19,18]	[12,17,20,19]	[12,17,20]
13	[13,17,16]	[13,17]	[13,17,20,19,18]	[13,17,20,19]	[13,17,20]
14	[14,13,17,16]	[14,13,17]	[14,13,17,20,19,18]	[14,13,17,20,19]	[14,13,17,20]
15	[15,16]	[15,16,17]	[15,18]	[15,16,19]	[15,16,19,20]
16	16	[16,17]	[16,19,18]	[16,19]	[16,19,20]
17	[17,16]	17	[17,20,19,18]	[17,20,19]	[17,20]
18	[18,19,16]	[18,19,20,17]	18	[18,19]	[18,19,20]
19	[19,16]	[19,20,17]	[19,18]	19	[19,20]
20	[20,19,16]	[20,17]	[20,19,18]	[20,19]	20

Table 9. The corresponding objective function values.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1	0	0	0	0	0	0	0	65.57	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	27.84	186.2	151.6	165.8	245.0	0	177.9	103.6	120.7	148.4	207.1	210.5	279.2	202.7	158.1	254.2	186.3	174.4
3	0	27.84	0	158.4	123.7	137.9	217.1	0	150.0	75.77	92.89	120.5	179.2	182.7	251.4	174.8	130.2	226.4	158.4	146.5
4	0	186.2	158.4	0	96.42	110.6	189.8	0	156.9	82.66	65.54	93.24	151.9	155.3	224.0	147.5	102.9	199.0	131.1	119.2
5	0	151.6	123.7	96.42	0	14.18	93.40	0	122.3	47.99	30.87	3.183	61.87	65.31	134.0	57.45	12.89	109.0	41.05	29.15
6	0	165.8	137.9	110.6	14.18	0	79.22	0	136.4	62.18	45.06	17.37	76.06	79.50	148.1	71.64	27.08	123.2	55.24	43.34
7	0	245.0	217.1	189.8	93.40	79.22	0	0	215.7	141.4	124.2	96.59	155.2	158.7	227.4	150.8	106.3	202.4	134.4	122.5
8	65.57	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	0	177.9	150.0	156.9	122.3	136.4	215.7	0	0	74.31	91.43	119.1	177.8	181.2	249.9	173.3	128.8	224.9	156.9	145.0
10	0	103.6	75.77	82.66	47.99	62.18	141.4	0	74.31	0	17.11	44.81	103.5	106.9	175.6	99.08	54.52	150.6	82.68	70.78
11	0	120.7	92.89	65.54	30.87	45.06	124.2	0	91.43	17.11	0	27.69	86.38	89.82	158.5	81.96	37.40	133.5	65.56	53.66
12	0	148.4	120.5	93.24	3.183	17.37	96.59	0	119.1	44.81	27.69	0	58.68	62.13	130.8	54.27	9.713	105.8	37.87	25.97
13	0	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	0	2e+6	2e+6	2e+6	2e+6	0	3.444	170.0	93.53	48.97	145.1	77.13	65.23
14	0	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	0	2e+6	2e+6	2e+6	2e+6	3.444	0	173.5	96.97	52.42	148.5	80.58	68.68
15	0	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	0	2e+6	2e+6	2e+6	2e+6	170.0	173.5	0	76.55	121.1	70.93	104.1	116.0
16	0	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	0	2e+6	2e+6	2e+6	2e+6	93.53	96.97	76.55	0	44.55	95.57	27.60	39.50
17	0	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	0	2e+6	2e+6	2e+6	2e+6	48.97	52.42	121.1	44.55	0	96.13	28.16	16.26
18	0	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	0	2e+6	2e+6	2e+6	2e+6	145.1	148.5	70.93	95.57	96.13	0	67.97	79.87
19	0	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	0	2e+6	2e+6	2e+6	2e+6	77.13	80.58	104.1	27.60	28.16	67.97	0	11.89
20	0	2e+6	2e+6	2e+6	2e+6	2e+6	2e+6	0	2e+6	2e+6	2e+6	2e+6	65.23	68.68	116.0	39.50	16.26	79.87	11.89	0

In this section, we critically examine the experimental results and provide a comprehensive discussion on the superior performance of iTEVAC over TEVAC. Figure 9(a–c) illustrate the evolving network of the commercial centre at three next temporal instances. These instances correspond to two specific scenarios: one prior to the elimination of paths with objective function values surpassing the threshold, and the other after this elimination.

Our empirical analysis and the ensuing discourse primarily concentrate on the comparative efficiency of the iTEVAC model versus the TEVAC model in the context of smart evacuation. The data unambiguously demonstrates that, at each juncture, the processing time required by iTEVAC is consistently less than that of TEVAC. Upon analyzing the data presented in Table 11 and Figure 10, it is evident that the iTEVAC model consistently outperforms the TEVAC model in terms of processing time across all considered time points.

At the initial time point (t1), the iTEVAC model demonstrates a processing time of approximately 1.44 s, a significant improvement over the TEVAC model's processing time of 1.97 s. This represents an efficiency increase of approximately 27%.

At the second time point (t2), the iTEVAC model requires only 0.35 s to process, compared to the TEVAC model's 2.24 s. This indicates that the iTEVAC model is approximately 84% faster than the TEVAC model at this stage.

This trend continues at the third time point (t3), where the iTEVAC model processes in approximately 0.37 s, while the TEVAC model requires about 2.07 s. This represents an efficiency increase of about 82% for the iTEVAC model.

At the final time point (t4), the iTEVAC model maintains its superior performance, requiring only about 0.32 s to process, compared to the TEVAC model's 2.15 s. This indicates an efficiency increase of about 85% for the iTEVAC model.

Table 11 presents the processing time of the proposed algorithm at four distinct time points in comparison to TEVAC. As the situation evolves, the processing time of the algorithm decreases. This is due to the rapid spread of the incident, which reduces the number of safe paths in the network, thereby reducing the processing load of the proposed algorithm.

This is a substantial leap forward in the realm of smart evacuation strategies. The significance of this improvement is amplified when one considers the criticality of time during emergency evacuations. The iTEVAC model, with its superior time efficiency,

Table 10. Title optimal evacuation paths for nodes to nearest exits.

Moving node	Optimal movement path	Direction and alert
2	[2,3]	Move to node 3
4	[4,11,10]	Move to node 11
5	[5,12]	Move to node 12
6	[6,5,12]	Move to node 5
8	[8,1]	Move to node 1
9	[9,10]	Move to node 10
11	[11,10]	Move to node 10
13	[13,17,12]	Move to node 17
14	[14,13,17,12]	Move to node 13
15	[15,16,17,12]	Move to node 16
16	[16,17,12]	Move to node 17
17	[17,12]	Move to node 12
18	[18,19,20,17,12]	Move to node 19
19	[19,20,17,12]	Move to node 20
20	[20,17,12]	Move to node 17

Table 11. Algorithm processing time in comparison between iTEVAC and TEVAC.

Time Point	Processing Time (seconds)	
	<i>iTEVAC</i>	<i>TEVAC</i>
t1	1.4414521	1.9657
t2	0.3509644	2.2393
t3	0.3719131	2.0694
t4	0.3177585	2.1549

underscores the importance of intelligent and swift decision-making processes in emergency management, thereby potentially saving lives and resources.

Table 12 displays the nodes that become isolated immediately after the removal of paths with objective function values exceeding the threshold at four different update times. The table also proposes solutions based on the output [1 3 7 10 12]. As the situation progresses, the number of isolated nodes increases, necessitating more extensive rescue efforts. Immediate intervention and emergency measures are essential to facilitate the evacuation of individuals at these nodes, ensuring their safety and security.

In addition, Tables 13–15 present guidance for movement and alerts for nodes [2, 4, 5, 6, 8, 9, 11, 13, 14, 15, 16, 17, 18, 19, 20], immediately after executing the safety optimization solution proposed by the algorithm at three next time points.

These tables provide a clear and structured representation of the optimal movement paths and corresponding directions and alerts for each node at different time points. This information is crucial for ensuring a smart and efficient evacuation process during emergencies. By having a well-organized set of instructions and alerts, the iTEVAC model can facilitate a more streamlined and effective evacuation, minimizing the time taken and ensuring the safety of individuals within the commercial centre or old buildings.

Furthermore, in the rapidly evolving field of emergency management, particularly in the context of evacuation procedures, the integration of advanced technologies has led to the development of various innovative systems. Each of these systems, designed with unique methodologies and technological frameworks, aims to enhance the safety and efficiency of evacuation processes in emergency scenarios. As part of our ongoing research and development in this domain, it is imperative to conduct a comprehensive comparative analysis of existing systems. This analysis not only benchmarks our system, iTEVAC, against other leading approaches but also provides valuable insights into the current state of technological advancements in emergency evacuation strategies.

The following table (Table 16) presents a detailed comparison of three distinct evacuation systems: the Integrated Evacuation and Rescue Management System (IERM) (Didar & Abbaspour, 2023), the BIM and Computer Vision-Based Framework (Deng et al., 2021), and our iTEVAC model. This comparison covers various dimensions including system design, methodology, simulation environment, key features, performance metrics, pros, and cons. This comprehensive analysis aims to provide a clear understanding of how iTEVAC stands in relation to these systems and to highlight its unique contributions to the field of emergency evacuation.

In concluding our comparative analysis of the Integrated Evacuation and Rescue Management System (IERM), the BIM and Computer Vision-Based Framework, and iTEVAC, we

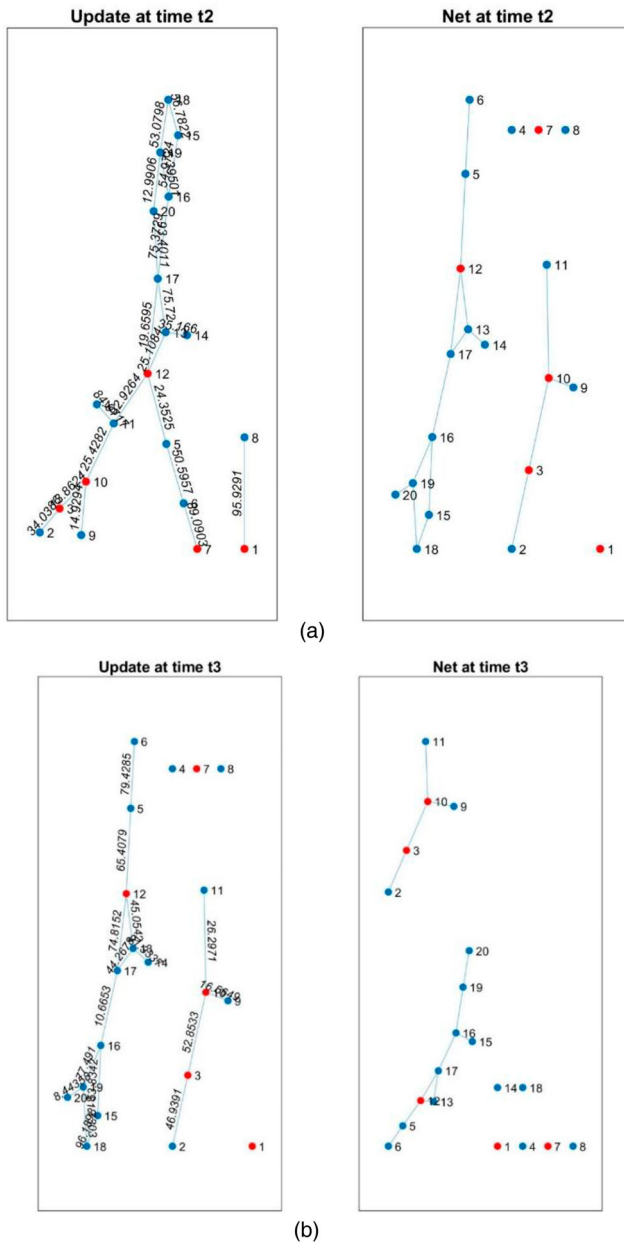


Figure 9. (a) Network at time t2. (b). Network at time t3. (c) Network at time t4.

gain critical insights into the landscape of emergency evacuation technologies. Each system presents a unique blend of technological integration and complexity, with iTEVAC’s rapid decision-making through Fog Computing and IoT standing out, albeit with a reliance on robust IoT infrastructure and connectivity. The comparison underscores the varying strengths and challenges of each system: IERM’s high success rate versus its implementation complexity, and the BIM-based framework’s precision in indoor

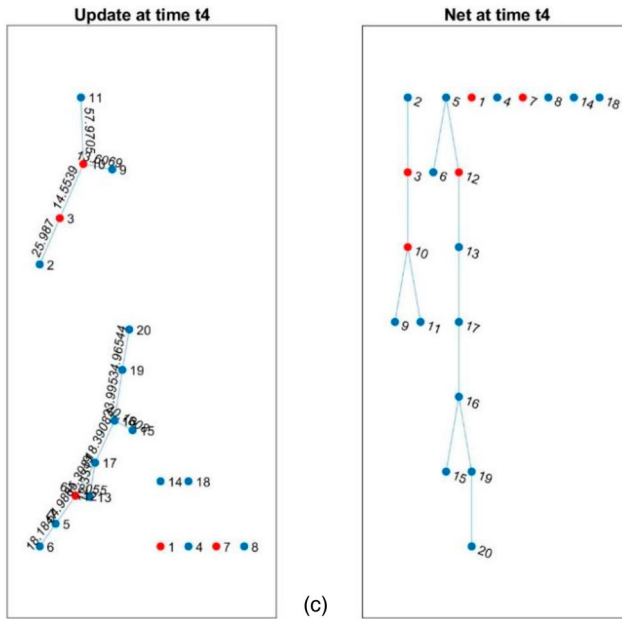


Figure 9 Continued

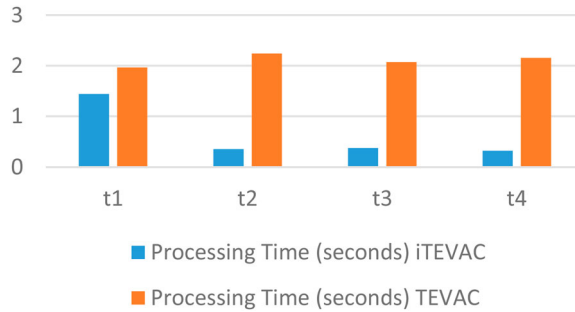


Figure 10. Comparison between iTEVAC and TEVAC.

positioning against its dependency on accurate BIM models. iTEVAC’s adaptability and potential scalability in diverse emergency scenarios are notable, suggesting its significant utility in complex environments. This analysis not only highlights areas for iTEVAC’s enhancement, particularly in bolstering its resilience to power and connectivity issues, but also suggests avenues for integrating features from other systems to enhance its capabilities. The rapid processing time and effective guidance offered by iTEVAC in urgent situations underscore its valuable contribution to emergency evacuation strategies. Moving forward, the insights from this study will be instrumental in refining iTEVAC, with a continued focus on enhancing safety, reducing response times, and effectively navigating the multifaceted challenges of emergency evacuations. This comparative study thus positions iTEVAC as a potentially leading solution in emergency management, poised for ongoing innovation and development.

Table 12. Isolated nodes and proposed solutions.

Time Point	Isolated nodes	Proposed solutions based on output [1 3 7 10 12]
t1	[0]	No rescue needed
t2	[1, 4, 7, 8]	Dispatch rescue to nodes 4–8 (1–7 are exits)
t3	[1, 4, 7, 8, 14, 18]	Dispatch rescue to nodes 4–8–14–18 (1–7 are exits)
t4	[1, 4, 7, 8, 14, 18]	Dispatch rescue to nodes 1–4–7–8–14–18

V. Work in progress

While the proposed iTEVAC model has demonstrated promising results in the context of smart evacuation within commercial centres, there are several avenues for future research that could further enhance its effectiveness and applicability.

A. Complexity and scalability

The current model has been tested primarily within the context of commercial centres. Future work could involve scaling the model to accommodate larger and more complex environments such as multi-building campuses, city districts, or even entire cities. This would require addressing challenges related to computational complexity and the integration of more diverse data sources.

B. Real-time data integration

The current iTEVAC model effectively utilizes IoT sensors within a Fog Computing environment to detect smoke, heat, and poisonous gases, providing real-time data that informs dynamic and responsive evacuation strategies. Moving forward, we aim to further enhance and expand these capabilities. Potential improvements could include refining the sensitivity and accuracy of the IoT sensors, developing more sophisticated algorithms for data analysis and decision-making, and expanding the range of detectable hazards. Additionally, we could explore the integration of other real-time data sources, such as CCTV feeds or social media updates, to provide a more comprehensive understanding of the situation on the ground. These enhancements would further improve the adaptability of our evacuation strategies to rapidly changing conditions, thereby increasing the safety and efficiency of emergency evacuations.

Table 13. Movement directions and alerts at time t2 for each node.

Moving node	Optimal movement path	Direction and alert
2	[2,3]	Move to node 3
4	[]	Call for rescue at node 4 as no exit path is found
5	[5,12]	Move to node 12
6	[6,5,12]	Move to node 5
8	[]	Call for rescue at node 8 as no exit path is found
9	[9,10]	Move to node 10
11	[11,10]	Move to node 10
13	[13,12]	Move to node 12
14	[14,13,12]	Move to node 13
15	[15,16,17,12]	Move to node 16
16	[16,17,12]	Move to node 17
17	[17,12]	Move to node 12
18	[18,19,16,17,12]	Move to node 19
19	[19,16,17,12]	Move to node 16
20	[20,19,16,17,12]	Move to node 19

Table 14. Movement directions and alerts at time t3 for each node.

Moving node	Optimal movement path	Direction and alert
2	[2,3]	Move to node 3
4	[]	Call for rescue at node 4 as no exit path is found
5	[5,12]	Move to node 12
6	[6,5,12]	Move to node 5
8	[]	Call for rescue at node 8 as no exit path is found
9	[9,10]	Move to node 10
11	[11,10]	Move to node 10
13	[13,12]	Move to node 12
14	[]	Call for rescue at node 14 as no exit path is found
15	[15,16,17,12]	Move to node 16
16	[16,17,12]	Move to node 17
17	[17,12]	Move to node 12
18	[]	Call for rescue at node 18 as no exit path is found
19	[19,16,17,12]	Move to node 16
20	[20,19,16,17,12]	Move to node 19

C. Human behavior modeling

The current model assumes a degree of rationality in the behaviour of individuals during an evacuation. Future work could incorporate more sophisticated models of human behaviour that account for factors such as panic, group dynamics, or individual physical capabilities.

D. Multi-objective optimization

The current model focuses on safety as the primary objective. Future iterations of the model could consider additional objectives such as evacuation speed, distance travelled, or the comfort and convenience of evacuees.

E. Validation and testing

Further empirical testing and validation of the model in real-world scenarios or through more extensive simulation studies would be beneficial. This could involve collaboration with emergency services or disaster management agencies.

Table 15. Movement directions and alerts at time t4 for each node.

Moving node	Optimal movement path	Direction and alert
2	[2,3]	Move to node 3
4	[]	Call for rescue at node 4 as no exit path is found
5	[5,12]	Move to node 12
6	[6,5,12]	Move to node 5
8	[]	Call for rescue at node 8 as no exit path is found
9	[9,10]	Move to node 10
11	[11,10]	Move to node 10
13	[13,12]	Move to node 12
14	[]	Call for rescue at node 14 as no exit path is found
15	[15,16,17,13,12]	Move to node 16
16	[16,17,13,12]	Move to node 17
17	[17,13,12]	Move to node 13
18	[]	Call for rescue at node 18 as no exit path is found
19	[19,16,17,13,12]	Move to node 16
20	[20,19,16,17,13,12]	Move to node 19

Table 16. Comparative analysis of evacuation systems.

Aspect Feature	Integrated evacuation and rescue management system (IERM)	BIM and computer vision-based framework	iTEVAC
System Design	Agent-based simulation in AnyLogic	BIM model, IFC JAVA TOOLBOX, MatlabR2018b	Fog Computing, IoT
Methodology	Dynamic evacuation and rescue with priority assessment	Indoor positioning and path planning with computer vision	Smart evacuation leveraging IoT and Fog Computing
Simulation Environment	Pedestrian library for people flows simulation	Three-floor campus building model for testing	Commercial centre, old building with a network of path-nodes and exit-nodes
Key Features	Real-time data integration, risk-aware evacuation, dynamic rescue planning	GNN generation, indoor local positioning, BP neural network for path planning	Intelligent navigation systems, real-time situational analysis, decision-making speed < 1 s
Performance Metrics	Evacuation success rate, rescue success rate, total response time	Positioning accuracy, safety performance of rescue routes	Processing time, efficiency in identifying safe evacuation routes
Pros	High success rate, efficient resource allocation, adaptable to various scenarios	High accuracy in indoor positioning, considers local safety in route planning	Rapid decision-making, effective in complex environments, scalable
Cons	Complexity in implementation, requires extensive data	Relies on accurate BIM models, complex setup for indoor positioning	May require extensive IoT infrastructure, dependent on power and system connectivity

F. Machine learning integration

Machine learning techniques could be incorporated to predict potential evacuation scenarios based on historical data, thereby improving the model’s predictive capabilities, and allowing for more proactive evacuation planning.

By pursuing these avenues of research, we can continue to improve upon the iTEVAC model and contribute to the development of more effective and intelligent evacuation strategies.

VI. Conclusion

The proposed algorithm has successfully executed alerts for isolated nodes and those unable to find an exit route, as well as provided detailed guidance for nodes that need to move within the network of the commercial centre. One of the key advantages of this algorithm is the decision-making time, which is less than 1 s. This aligns with the real-world scenario where, immediately after an incident, there is a high probability of power failure, system disconnection, or unforeseen events affecting the entire area. Therefore, providing guidance within 1 s before a power outage is a valuable strategy for saving human lives. We acknowledge that no solution is perfect, and our aim is to propose strategies that minimize human risk.

In the context of smart evacuation, this conclusion emphasizes the efficiency and effectiveness of the proposed algorithm in a high-stakes, time-sensitive situation. The algorithm’s ability to quickly identify isolated nodes and provide immediate guidance for safe evacuation routes is a significant advancement in emergency response strategies. The decision-making speed of less than 1 s is particularly crucial, as it allows for immediate action in the face of potential power loss or system disconnection, thereby maximizing the chances of successful evacuation and minimizing the risk to human lives.

Disclosure statement

No potential conflict of interest was reported by the author(s).

Funding

This work belongs to the project 2024 funded by Ho Chi Minh City University of Technology and Education, Ho Chi Minh City, Vietnam.

ORCID

Thinh Vinh Le  <http://orcid.org/0000-0001-5951-096X>

References

- Al Shereiqi, I. M., & Sohail, M. M. (2020). Smart fire alarm system using IOT. *Journal of Student Research*, <https://doi.org/10.47611/jsr.vi.882>
- Al-Nabhan, N., Al-Aboody, N., & Alim Al Islam, A. B. M. (2019). A hybrid IoT-based approach for emergency evacuation. *Computer Networks*, *155*, 87–97. <https://doi.org/10.1016/j.comnet.2019.03.015>
- Araujo Lima, G. P., Viana Barbosa, J. D., Beal, V. E., Moret S. Gonçalves, M. A., Souza Machado, B. A., Gerber, J. Z., & Lazarus, B. S. (2021). Exploratory analysis of fire statistical data and prospective study applied to security and protection systems. *International Journal of Disaster Risk Reduction*, *61*, 102308. <https://doi.org/10.1016/j.ijdr.2021.102308>
- Atila, U., Ortakci, Y., Ozacar, K., Demiral, E., & Karas, I. (2018). Smartescape: A mobile smart individual fire evacuation system based on 3D spatial model. *ISPRS International Journal of Geo-Information*, *7*(6), 223. <https://doi.org/10.3390/ijgi7060223>
- Bhawesh, C., & Gouthamkumar, N. (2020). Advanced integrated fire controlling system to reduce casualty and minimize deaths. In *2020 2nd international conference on advances in computing, communication control and networking (ICACCCN)* (pp. 472–476). <https://doi.org/10.1109/ICACCCN51052.2020.9362770>
- Chen, A. Y., & Chu, J. C. (2016). Tdvrp and BIM integrated approach for in-building emergency rescue routing. *Journal of Computing in Civil Engineering*, *30*(5), C4015003. [https://doi.org/10.1061/\(ASCE\)CP.1943-5487.0000522](https://doi.org/10.1061/(ASCE)CP.1943-5487.0000522)
- Cheng, M.-Y., Chiu, K.-C., Hsieh, Y.-M., Yang, I.-T., & Chou, J.-S. (2016, July 21). Development of BIM-based real-time evacuation and rescue system for complex buildings. In *33th international symposium on automation and robotics in construction*, Auburn, AL, USA. <https://doi.org/10.22260/ISARC2016/0120>.
- Cheng, M.-Y., Chiu, K.-C., Hsieh, Y.-M., Yang, I.-T., Chou, J.-S., & Wu, Y.-W. (2017). BIM integrated smart monitoring technique for building fire prevention and disaster relief. *Automation in Construction*, *84*, 14–30. <https://doi.org/10.1016/j.autcon.2017.08.027>
- Cheung, W.-F., Lin, T.-H., & Lin, Y.-C. (2018). A real-time construction safety monitoring system for hazardous gas integrating wireless sensor network and building information modeling technologies. *Sensors*, *18*(2), 436. <https://doi.org/10.3390/s18020436>
- Cortés-Pérez, J. P., Cortés-Pérez, A., & Prieto-Muriel, P. (2020). BIM-integrated management of occupational hazards in building construction and maintenance. *Automation in Construction*, *113*, 103115. <https://doi.org/10.1016/j.autcon.2020.103115>
- Deng, H., Ou, Z., Zhang, G., Deng, Y., & Tian, M. (2021). Bim and computer vision-based framework for fire emergency evacuation considering local safety performance. *Sensors*, *21*(11), 3851. <https://doi.org/10.3390/s21113851>
- Didar, N., & Abbaspour, M. (2023). Integrated evacuation and rescue management system in response to fire incidents. *European Journal of Engineering and Technology Research*, *8*(1), 1–12. <https://doi.org/10.24018/ejeng.2023.8.1.2946>

- Fang, H., Lo, S., & Lo, J. T. Y. (2021). Building fire evacuation: An IoT-aided perspective in the 5G Era. *Buildings*, 11(12), 643. <https://doi.org/10.3390/buildings11120643>
- Fu, M., & Liu, R. (2020). BIM-based automated determination of exit sign direction for intelligent building sign systems. *Automation in Construction*, 120, 103353. <https://doi.org/10.1016/j.autcon.2020.103353>
- Gaur, A., Singh, A., Verma, A., & Kumar, A. (2022). Artificial intelligence and multi-sensor fusion based universal fire detection system for smart buildings using IoT techniques. *IETE Journal of Research*, 1–13. <https://doi.org/10.1080/03772063.2022.2088626>
- Gelenbe, E., & Wu, F.-J. (2012). Large scale simulation for human evacuation and rescue. *Computers & Mathematics with Applications*, 64(12), 3869–3880. <https://doi.org/10.1016/j.camwa.2012.03.056>
- Ibrahim, A. M., Venkat, I., Subramanian, K. G., Khader, A. T., & Wilde, P. D. (2016). Intelligent evacuation management systems: A review. *ACM Transactions on Intelligent Systems and Technology*, 7(3), 1–27. <https://doi.org/10.1145/2842630>
- Immediately Dangerous To Life or Health (IDLH) Values. (n.d.). <https://www.cdc.gov/niosh/idlh/intridl4.html>
- Johnston, M. J., King, D., Arora, S., Behar, N., Athanasiou, T., Sevdalis, N., & Darzi, A. (2015). Smartphones let surgeons know WhatsApp: An analysis of communication in emergency surgical teams. *The American Journal of Surgery*, 209(1), 45–51. <https://doi.org/10.1016/j.amjsurg.2014.08.030>
- Khan, A., Aesha, A. A., Aka, J. S., Rahman, S. M. F., & Rahman, M. J.-U. (2018). An IoT based intelligent fire evacuation system. In *2018 21st International conference of computer and information technology (ICCIIT)* (pp. 1–6). <https://doi.org/10.1109/ICCICTECHN.2018.8631945>
- Kodali, R. K., & Yerroju, S. (2017). IoT based smart emergency response system for fire hazards. In *2017 3rd International conference on applied and theoretical computing and communication technology (iCATccT)* (pp. 194–199). <https://doi.org/10.1109/ICATCCCT.2017.8389132>
- Le, T. V., & Huan, T. T. (2020). Computational intelligence towards trusted cloudlet based Fog computing. In *2020 5th International conference on green technology and sustainable development (GTSD)* (pp. 141–147). <https://doi.org/10.1109/GTSD50082.2020.9303091>
- Levin, S., Toerper, M., Hamrock, E., Hinson, J. S., Barnes, S., Gardner, H., Dugas, A., Linton, B., Kirsch, T., & Kelen, G. (2018). Machine-learning-based electronic triage more accurately differentiates patients with respect to clinical outcomes compared with the emergency severity index. *Annals of Emergency Medicine*, 71(5), 565–574.e2. <https://doi.org/10.1016/j.annemergmed.2017.08.005>
- Li, B., Li, T., & Huang, D. (2018). Research on automatic fire monitoring and recognition technology based on digital image processing technology. In *Proceedings of the 2018 8th International conference on management, education and information (MEICI 2018)*, Shenzang, People's Republic of China. <https://doi.org/10.2991/meici-18.2018.16>
- Li, N., Becerik-Gerber, B., Krishnamachari, B., & Soibelman, L. (2014). A BIM centered indoor localization algorithm to support building fire emergency response operations. *Automation in Construction*, 42, 78–89. <https://doi.org/10.1016/j.autcon.2014.02.019>
- Li, N., Yang, Z., Ghahramani, A., Becerik-Gerber, B., & Soibelman, L. (2014). Situational awareness for supporting building fire emergency response: Information needs, information sources, and implementation requirements. *Fire Safety Journal*, 63, 17–28. <https://doi.org/10.1016/j.firesaf.2013.11.010>
- Lurz, F., Mueller, S., Lindner, S., Linz, S., Gardill, M., Weigel, R., & Koelpin, A. (2017). Smart communication and relative localization system for firefighters and rescuers. In *2017 IEEE MTT-S International microwave symposium (IMS)* (pp. 1421–1424). <https://doi.org/10.1109/MWSYM.2017.8058885>
- Ma, G., & Wu, Z. (2020). BIM-based building fire emergency management: Combining building users' behavior decisions. *Automation in Construction*, 109, 102975. <https://doi.org/10.1016/j.autcon.2019.102975>
- Mahmud, R., Kotagiri, R., & Buyya, R. (2018). *Fog computing: A taxonomy, survey and future directions* (pp. 103–130). https://doi.org/10.1007/978-981-10-5861-5_5
- Mehra, S., & Tanmay. (2017). Disaster detection system using arduino. In *2017 International conference on information communication and embedded systems (ICICES)* (pp. 1–6). <https://doi.org/10.1109/ICICES.2017.8070777>

- Mirahadi, F., McCabe, B., & Shahi, A. (2019). IFC-centric performance-based evaluation of building evacuations using fire dynamics simulation and agent-based modeling. *Automation in Construction*, 101, 1–16. <https://doi.org/10.1016/j.autcon.2019.01.007>
- Mohammadiounotikandi, A., FakhruLdeen, H. F., Meqdad, M. N., Ibrahim, B. F., Jafari Navimipour, N., & Unal, M. (2023). A fire evacuation and control system in smart buildings based on the internet of things and a hybrid intelligent algorithm. *Fire*, 6(4), 171. <https://doi.org/10.3390/fire6040171>
- Nguyen, V.-Q., Vu, H.-T., Nguyen, V.-H., & Kim, K. (2022). A smart evacuation guidance system for large buildings. *Electronics*, 11(18), 2938. <https://doi.org/10.3390/electronics11182938>
- Nikoloudakis, Y., Panagiotakis, S., Markakis, E., Pallis, E., Mastorakis, G., Mavromoustakis, C. X., & Dobre, C. (2016). A Fog-based emergency system for smart enhanced living environments. *IEEE Cloud Computing*, 3(6), 54–62. <https://doi.org/10.1109/MCC.2016.118>
- Nowak, T., Hartmann, M., Troger, H.-M., Patino-Studencki, L., & Thielecke, J. (2017). Probabilistic multipath mitigation in RSSI-based direction-of-arrival estimation. In *2017 IEEE International conference on communications workshops (ICC workshops)* (pp. 1024–1029). <https://doi.org/10.1109/ICCW.2017.7962793>
- Okokpujie, K., John, S. N., & Noma-Osaghae, E. (n.d.). *A wireless sensor network based fire protection system with SMS alerts*.
- Ortiz-Garcés, I., Andrade, R. O., Sanchez-Viteri, S., & Villegas-Ch, W. (2023). Prototype of an emergency response system using IoT in a Fog computing environment. *Computers*, 12(4), 81. <https://doi.org/10.3390/computers12040081>
- OSHA Occupational Chemical Database-Documentation of the Threshold Limit Values (TLVs) and Biological Exposure Indices (BEIs)—Carbon Dioxide. (n.d.). <https://www.osha.gov/chemicaldata/183>
- Park, S., Park, S., Park, L., Park, S., Lee, S., Lee, T., Lee, S., Jang, H., Kim, S., Chang, H., & Park, S. (2018). Design and implementation of a smart IoT based building and town disaster management system in smart city infrastructure. *Applied Sciences*, 8(11), 2239. <https://doi.org/10.3390/app8112239>
- Raita, Y., Goto, T., Faridi, M. K., Brown, D. F. M., Camargo, C. A., & Hasegawa, K. (2019). Emergency department triage prediction of clinical outcomes using machine learning models. *Critical Care*, 23(1), 64. <https://doi.org/10.1186/s13054-019-2351-7>
- Rashid, K. M., Louis, J., & Fiawoyife, K. K. (2019). Wireless electric appliance control for smart buildings using indoor location tracking and BIM-based virtual environments. *Automation in Construction*, 101, 48–58. <https://doi.org/10.1016/j.autcon.2019.01.005>
- Sahid, D., & Alaydrus, M. (2020). Multi sensor fire detection in low voltage electrical panel using modular fuzzy logic. In *2020 2nd International conference on broadband communications, wireless sensors and powering (BCWSP)* (pp. 31–35). <https://doi.org/10.1109/BCWSP50066.2020.9249400>
- Sahil, & Sood, S. K. (2022). Fog-cloud centric IoT-based cyber physical framework for panic oriented disaster evacuation in smart cities. *Earth Science Informatics*, 15(3), 1449–1470. <https://doi.org/10.1007/s12145-020-00481-6>
- Sloot, R. N. F., Heutink, A., & Voordijk, J. T. (2019). Assessing usefulness of 4D BIM tools in risk mitigation strategies. *Automation in Construction*, 106, 102881. <https://doi.org/10.1016/j.autcon.2019.102881>
- Tang, Y., Xia, N., Lu, Y., Varga, L., Li, Q., Chen, G., & Luo, J. (2021). BIM-based safety design for emergency evacuation of metro stations. *Automation in Construction*, 123, 103511. <https://doi.org/10.1016/j.autcon.2020.103511>
- Vinh, T. L., Thien, H. T., Huu, T. N., & Bouzefrane, S. (2023). TEVAC: Trusted evacuation system based Fog computing. In *2023 international conference on system science and engineering (ICSSE)* (pp. 164–169). <https://doi.org/10.1109/ICSSE58758.2023.10227231>
- Wang, J., Wei, G., & Dong, X. (2021). A dynamic fire escape path planning method with BIM. *Journal of Ambient Intelligence and Humanized Computing*, 12(11), 10253–10265. <https://doi.org/10.1007/s12652-020-02794-2>
- Wu, X., Zhang, X., Huang, X., Xia, F., & Usmani, A. (2022). A real-time forecast of tunnel fire based on numerical database and artificial intelligence. *Building Simulation*, 15(4), 511–524. <https://doi.org/10.1007/s12273-021-0775-x>

- Yousefpour, A., Fung, C., Nguyen, T., Kadiyala, K., Jalali, F., Niakanlahiji, A., Kong, J., & Jue, J. P. (2019). All one needs to know about Fog computing and related edge computing paradigms: A complete survey. *Journal of Systems Architecture*, 98, 289–330. <https://doi.org/10.1016/j.sysarc.2019.02.009>
- Zuolkernan, I. A., Aloul, F. A., Sakkia, V., Noman, H. A., Sowdagar, S., & Hammadi, O. A. (2019). An IoT-based emergency evacuation system. In *2019 IEEE International conference on internet of things and intelligence system (IoTals)* (pp. 62–66). <https://doi.org/10.1109/IoTals47347.2019.8980381>