A Hybrid Emergency Evacuation Approach Using IoT

Amol L. Deokate[1], Yogesh A. Pawar[2], Pravin M. Dhanrao[3]
[1, 2, 3] Dept of Information Tech
[1, 2, 3] Sanjivani KBP College

Abstract- *This paper proposes a hybrid emergency evacuation approach using IoT and combines it with cloud computing. Internet of Things (IoT) offer solutions for utilization of the available resources. It has have potential to provide safe, reliable, and efficient solutions. This proposed approach is to utilize the advantages of both infrastructures like IoT and cloud to calculate the evacuation path in real time during emergency evacuation. It maximize the safety of the suggested evacuation paths by adapting to the characteristics of the hazard, evacuees' behavior, and environmental conditions. Our approach covers A Localized Emergency Navigator and a High- Risk Emergency Navigator. Depending on emergency factors, our approach decides which navigator is to be executed. It handles diversified evacuation issues, like when people get locked in a safe, a dead end area of a building under emergency conditions. This proposed approach achieves better outcomes in terms of its survival rate and evacuation efficiency.*

Keywords- Internet of Things, cloud computing, Applications, Emergency evacuation, Indoor navigation

I. INTRODUCTION

In emergency situation or condition that causes a hazard to an environment, life, community, or property. Emergency management (EM) is vital for any organization today. It aims to create plans by which communities reduce their vulnerability to hazards and cope with disasters. Generally speaking, coping with a threat or an incident includes three strategies: (1) controlling the threatening event; (2) controlling human settlement patterns; and/or (3) development of forecasting techniques and warning systems that generate a protective response to those threats [1].

Today's infrastructure and building management applications include several heterogeneous wireless sensor networks (WSNs) and IoT-based applications for monitoring structural health, waste, air quality, noise, energy consumption, traffic, emergency, smart services such as smart parking and lighting, etc. WSNs are capable of providing digital interfaces to real-world things. WSNs are integrated recently with other communication and intelligent technologies, such as IoT, cloud computing, smartphones, and robots, in order to implement systems with more powerful, advanced, and accurate solutions. At the level of "things", devices (sensor nodes) perform sensing and communication over the network for delivering data to the sink or the gateway. The gateway forwards collected data to remotely located servers through the Internet. This provides feasibility for database storage, maintenance and data processing (e.g., analytics) to facilitate the user with a great interface and ubiquitous connectivity [2–4].

Meanwhile, adopting IoT for emergency management is considered to be promising from different perspectives: (1) the exist-ing heterogeneous and geographically distributed safety-related re-sources. These resources can be easily upgraded to be IoT devices by incorporating sensing and communication capabilities; (2) IoT enables easier access and interaction with existing wide variety of devices such as home appliances, surveillance cameras, monitoring sensors, actuators, displays, vehicles, and so on; (3) IoT encourages the development of a number of applications that make use of enormous and heterogeneous data generated by such objects to provide new services to citizens, companies, and public administrations. Indeed, IoT paradigm finds applications in many different domains, such as smart buildings and cities, smart health care, smart transportation, industrial automation, and smart grids. (4) the growing population increases the pressure on various aspects of urban life, and increases in the number and intensity of disasters (e.g., earthquakes, fires, floods, terrorist attacks or incidents), which accordingly increased interest in emerging technology for emergency situations and management to reduce the possibility of serious human casualties and property damage

The trade-off between centralized decisions made remotely by the cloud and localized distributed decisions made by local re-sources (i.e. sensor nodes) is important. Centralized decisions are generally expensive in terms of time and communication costs. However, the centralized decisions are important in some situation to minimize damage and fatalities, especially when localized decisions lack in making proper and accurate evacuation decisions. Such trade-off is affected by many factors, including timing, intensity (or perception) of the hazard, evacuees' behavior, and environmental conditions, all of which are considered in tackling time-critical evacuation tasks.

In this paper, we propose a hybrid approach for dynamic emergency evacuation using IoT and WSNs. It utilizes different types of sensor nodes to perform sensing and monitoring, make a localized decision during certain evacuation conditions, and dynamically control actuators (path signs) during the evacuation. It also utilizes the powerful communication and computation capabilities of IoT cloud in performing more sensitive evacuation decisions centrally, communicating with external evacuation authorities, and integrating with the localized WSN evacuation tasks to avoid casualties and complex evacuation situations caused by inaccuracy of localized evacuation approaches (such as dead ends).

II. RELATED WORK

The investigation of emergency management and navigation was previously motivated by defense applications. Emergency management has attracted many researchers in recent years as a result of increasing threats and unpredictable sources and types of hazards. Accordingly, several approaches and models have been proposed in the literature to autonomously act during a hazard to improve evacuation efficiency. This section provides an overview of the existing emergency navigation approaches in the literature.

Generally speaking, most of the available emergency systems rely on a static scene in the calculation of safest evacuation paths and they are performed separately from path-finding for rescue teams. Reference [12] proposes a state-of-the-art dynamic approach, which deals with a 3D environment, hazard locations, and dynamic distribution of occupants during the evacuation. A database of densities and information about hazard influence are generated and used to calculate optimal paths for rescue teams. Their findings revealed that static simulation is significantly different from semi-dynamic and dynamic simulations. These results have significant implications in achieving a rapid and safe evacuation of people during an emergency event.

A recent methodology for optimizing variable pedestrian evacuation guidance in buildings with convex polygonal interior spaces was proposed in [14]. It has three major contributions including

1) Calibrating a logistic regression model for guidance compliance behavior using a virtual reality experiment and identifying the critical factors for the behavior. 2) Considering the guidance compliance and following behaviors in the lower-level problem. 3) Calculating benchmarks to evaluate the performance of optimized variable guidance. The proposed methodology was validated with numerical examples. Results show that the method has a potential to reduce evacuation time in emergencies.

Another approach for smart building evacuation was proposed in [3] using cognitive packet networks (CPNs) with time and distance metrics. This approach aims to evacuate people based on their health conditions. A simulation was conducted using 30 evacuees. The results showed that this approach outperformed the Dijkstra-based evacuation algorithm in terms of time and survival rate. A novel approach for managing crowds in hazards using dynamic grouping was proposed in [1]. It enhances survival rates through implementing a quality of service (QoS) driven routing algorithm to handle different types of evacuees based on their age and health condition [1].

Another recent indoor localization solution for evacuation management in emergency scenarios was proposed in [16]. It presents a comprehensive system for the efficient management and monitoring of workers' evacuation in the context of factories and office buildings. The system is composed of a central monitoring unit and an application running on the mobile device of the user. Extensive simulation study is yet to be conducted to test the efficiency of the developed systems. Similar to the above approach, this study is yet to conduct different pragmatic emergency aspects.

A number of bio-inspired approaches for emergency evacuation were proposed in the literature. In [17], a new path planning approach is proposed for emergency evacuation that combines the Extended Social Force Model (ESFM) and the Improved Artificial Bee Colony (IABC) algorithm to enhance the visual realism and improve the efficiency of crowd evacuation. The proposed IABC algorithm improves the evacuation efficiency and provides sup-port for building design and evacuation management by employing the strategies of grouping and exit selection. It uses the evacuation time of the individuals as an evaluation metric. If an exit is overcrowded and congested, the individuals assess the degree of congestion, estimate the time needed to escape, and determine whether to select a farther exit for escape. Simulation results show the effectiveness of their method.
The presented approach in this paper aims to eliminate the shortcomings of the above approaches in that it employs an intelligent WSN-based navigator that autonomously calculates evacuation paths and acts as a first-response, real-time system. In addition, it utilizes the computation and communication capabilities of IoT cloud to make higher-order decisions that are difficult to be calculated locally using simple sensor nodes. We extended and used our previously implemented event-driven simulator, presented in [19], to simulate and evaluate the performance of the proposed approach. Note that, the

presented approach in this paper differs from other recent and related studies that focus on minimizing evacuation time but ignore other important safety aspects. Additionally, unlike many other existing similar approaches [1,3,11–21], our approach focuses on creating and studying the balance between the localized and centralized processing of evacuation decisions. It employs the proposed design in detecting and handling a critical evacuation problem when people are guided to dead-end areas of the building. Our approach is intended to be realistic in the sense that it does not require any prior configuration or function by evacuees during a hazard compared to the existing cloud-based evacuation approaches. Moreover, it is more practical in the sense that it does not rely on specific applications or users' smartphones in localizing and evacuating pedestrians. Nonetheless, our approach realizes a delicate blending of distances to exit(s) and incident, number of exit(s), risk, and hazard intensity, which is yet to be focused in the literature to the best of our knowledge. Such a realization results in substantially increased survival rate experiencing a marginal degradation in evacuation time. We present the detailed design and assumptions of our proposed approach in the next section.

III. SYSTEM ARCHITECTURE DESIGN

In this section, we present the conceptual model of the pro-posed approach. We first model the underlying network. Then, we present our assumptions related to the evacuation area.

3.1. Modeling the underlying network

Fig. 1 shows our underlying network diagram which integrates WSNs with IoT infrastructure for emergency and evacuation management. It mainly consists of: wireless sensor nodes, one local gateway (LG), IoT cloud which includes IoT servers, and a gateway that connects IoT cloud to the service plane where authorities such as country agencies (national security), city agencies (police, ambulance, fire engine), and community and web services are located.

The figure shows that WSNs act on the base plane, where low-cost sensing nodes (SNs) are densely deployed in the targeted area. SNs collect and transmit data to their nearby control or decision nodes (DNs). DNs are more capable than SNs, as they have higher computation and communication capabilities. However, in our model, DNs are deployed less-intensely than sensor nodes to minimize the cost and communication overhead. DNs communicate to IoT cloud through the home (local) gateway to tackle complex computations or provide remote information. On the other side of IoT cloud, it is connected to the service (or control) plane through another gateway(s).

SNs are nodes of different sensing capabilities used to detect the hazard (e.g., fire and gas) by measuring specific parameters (such as air quality, CO_2, and/or temperature). These nodes also assumed to have the ability to detect the presence of evacuees in their vicinity. In other words, these nodes present a combination of hazard and motion sensing. SNs communicate with their neigh-boring decision nodes, using short-range wireless communication, to transmit the sensed data.

DNs act as routers that execute the Localized Emergency Navigator (LEN) approach in order to calculate the best evacuation path to guide the evacuees in the nearby area. DNs are also able to interact with the LG. For path signs, we make an assumption that DNs are connected to installed actuator or path signs (i.e., LCDs) in order to dynamically show the calculated evacuation directions to the evacuees.

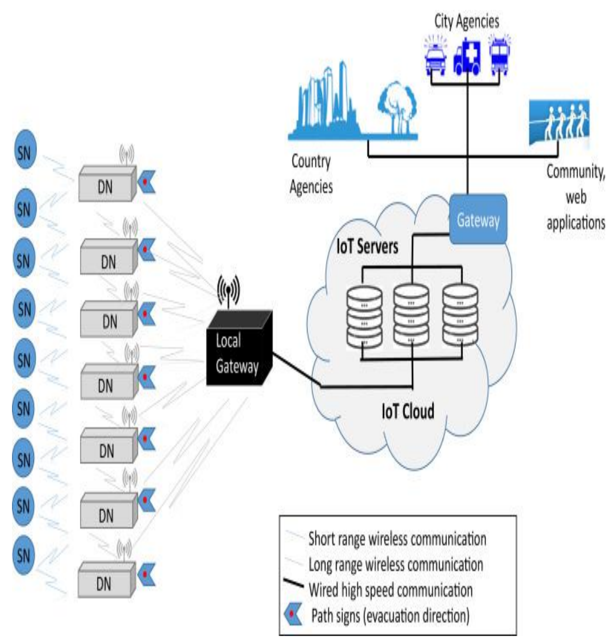

Fig. 1. Network diagram.

SNs and DNs can be connected to each other's and to the LG through an appropriate wireless protocol, such as ZigBee, Wi-Fi, or Bluetooth. These wireless protocols are the backbone of IoT systems. ZigBee is a low energy consumption and low-cost devices. It supports wireless mesh networking and can cover longer ranges. Bluetooth also consumes low energy while it depends heavily on its central unit and supports a very limited number of nodes. Wi-Fi requires nearby infrastructure to collect the data and consumes higher power than ZigBee and Bluetooth. Enabling IoT applications require Wi-Fi, Ethernet or 4 G connection.

Communication with IoT cloud is done through LG using high speed and reliable communication. The collected data is transferred to the IoT cloud to perform more advanced processing and analysis that requires higher computation capabilities. Communications also occur in specific evacuation conditions, especially in case of safe, however dead-end areas. DNs can also communicate with the IoT cloud directly using 6Lowpan protocol, which allows small devices with limited processing ability to transmit information using wireless communication to the internet.

It is worth mentioning that, since forming a response team and assigning responsibilities is one of the crucial steps in emergency response planning, monitoring data collection, and IoT cloud management is expected to be a responsibility assigned to the organization's safety unit. The safety unit is generally represented by the chief of security, safety shift supervisor, technical team, and/or incident commanders. The safety unit is expected to be the manager of the infrastructure, who would utilize sensors and perform processing of acquiring data from the cloud. Besides, the personnel working in the environment having the infrastructure would be the ultimate beneficiary in case there happens any fire hazard that exploits the infrastructure for the purpose of evacuation of the personnel working at the time of hazard.

3.2. Modeling the evacuation area

To model the underlying evacuation area, we use a similar building model as presented in the study [1]. Fig. 2 shows an exemplary model for the underlying area, which corresponds to the bottom-most plane in Fig. 1. The evacuation area is divided into zones covered by SNs and DNs. Here, SNs and DNs are deployed with two different alignment distances (*Stepx, Stepy*) and (*Stepdx, Stepdy*), respectively. Each zone is covered by four DNs and at least four SNs to provide full sensing coverage for the building. The gray lines (grid) represent walls that block evacuees' movements from one area (zone or room) to another. Moving from one zone to an-other should be through the specified door of the zone. Two exits are adapted in this exemplary model, one at the top right corner and the other at the bottom left corner of the area. We studied the impact of different operational parameters pertinent to the evacuation area.

IV. THE PROPOSED APPROACHES

4.1. A localized emergency navigator

Fig. 3 presents a conceptual design model for the overall behavior of the localized emergency navigator (LEN). LEN is triggered when a hazard is detected. SNs periodically collect and re-port data on hazard source, intensity, and evacuees' movements to the DNs. Consequently, DNs gather and combine data received from the nearby SNs with information provided by nearby DNs, LG and IoT cloud. Then, DNs employ LEN to locally calculate the appropriate evacuation paths.

The calculation of evacuation paths at DNs is done in a distributed manner through the following steps:

Step 1. At time t, each DN (d_i) evaluates its nearby paths in order to calculate its safety metric, $S_{(i,t)}$, as follows:

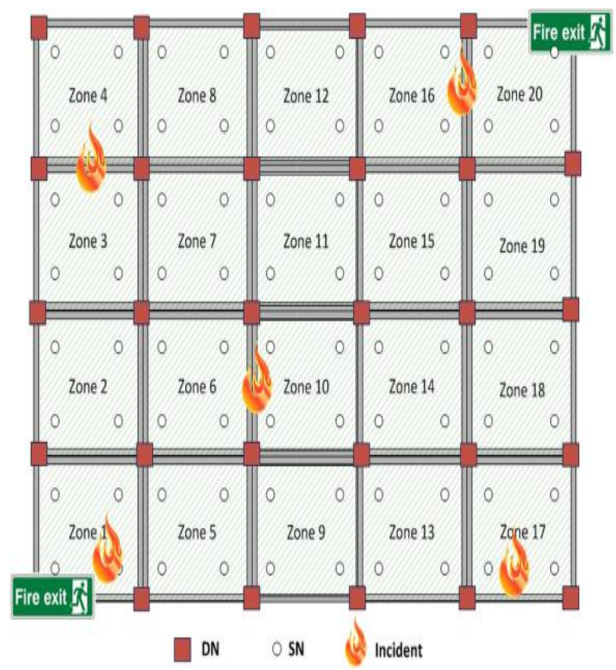

Fig. 2. A Graphical representation of an exemplary building model where the proposed approach can be employed

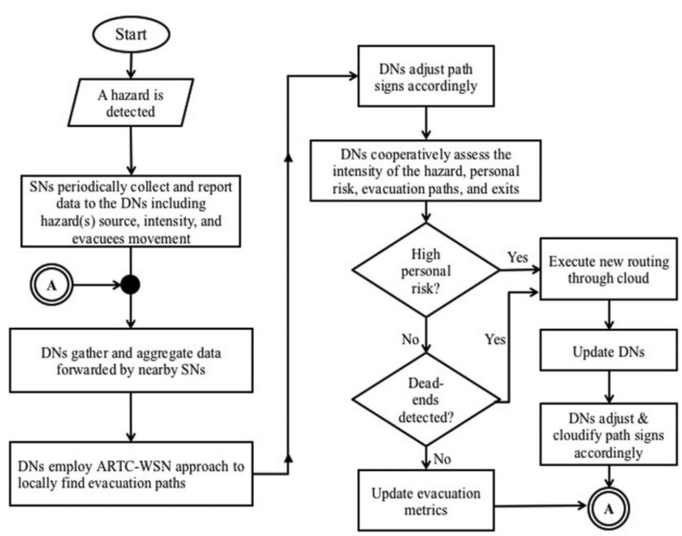

Fig. 3. The conceptual model representing the overall behavior of the proposed approach.

$WayOutIndicator(d_i) = distance(d_i, EX) \times Exit\ Factor$

$$1) \qquad S_{(i,t)} = \frac{RiskIndicator(i)}{WayOutIndicator(i)}$$

The exit factor in the above equation is a scalar number Within the range [1,2] and is calculated as $(1 + {}^{1}/\ no.\ of\ exits)$

$RiskIndicator(d_i) = distance(d_i, Incident) \times Risk\ Factor \times Intensity$ 2) *Distance*(d_i, *EX*) represents the distance between the decision node

d_i and its nearest exit. *Distance* (d_i, *Incident*) represents the distance between d_i and the location of incident happening.

The intensity in Eq. (2) corresponds to the spreading out of the incident over time between any point in time *t* and its Previous point *t-1*.

Step 2. Each decision node d_i exchanges its safety metric $S_{(i,t)}$ with its neighboring decision nodes.

Step 3. Each decision node d_i finds its best neighboring decision node, say d_j, among its neighbors by comparing its safety metric S_i with the safety metric of all neighboring decision nodes. The best neighboring decision node d_j of any node d_i is the one that has the highest safety metric, including the node d_i itself.

Step 4. At time *t*, each decision node d_i adjusts its controllable path signs to point toward the best decision node d_j. In addition, each decision node cooperatively assesses the intensity of the hazard, personal risk, evacuation paths, and exists with its neighbor.

Step 5. If high personal risk is detected, meaning that the incident is very close and/or has reached a nearby area where an evacuee is present, decision node d_i communicates with the cloud to invoke HREN navigator and updates evacuation parameters.

If a dead-end area is detected, decision node d_i communicates with the cloud to perform the routing task.

4.2. A high-risk emergency navigator

In our model, all DNs periodically report important information about the hazard and progress of evacuation to IoT cloud through LG. When high personal risk is detected, either by IoT cloud or locally by a DN, IoT cloud executes HREN to adjust evacuation metrics at specific DNs and/or acts to rescue people in from in the area.

IoT cloud also plays another important role in evacuating people from safe dead-end areas detected at any decision node d_i. *Safe dead-end conditions occur when* evacuees cannot be moved from their current location, toward the exit because all other nearby areas are considered by the surrounding DNs to be more dangerous than their current locations.

In this situation, the *cloud server* executes a different navigation mechanism, called HREN, that uses reverse routing. HREN attempts to find a route from the safest exit to the dead-end point, where evacuees are located, in order to find the best (shortest and safest) evacuation path for people in such areas. Reverse path finding also matches the direction of external rescue initiated by country agencies and community towards the location of the evacuees, which can help in saving their life and reaching them through the safest and shortest reverse route. In summary, HREN reverse routing aims at provide a fast and greedy solution by executing the following steps:

Step 1: When a dead-end point is detected by a DN d_i, it communicates to the IoT cloud to request help in updating evacuation metrics and executing reverse routing.

Step 2: IoT cloud locates the nearest exit to the dead-end area with the highest possible safety metric, and uses it as a starting point for the reverse routing.

Step 3: Given the safety metric S_j and location of all DNs j, the cloud selects the DN that is closer to the located dead-end area and has the highest possible safety among the other alternatives as a next hop.

Step 4: HREN keeps selecting the safest next-hop towards the dead-end.

Step 5: It terminates when the dead-end area is reached. Accordingly, all DNs along this path adjust the path signs based on the calculated reverse path, that provides the highest possible safety.

An important characteristic provided by HREN is that it gives higher importance to evacuation information received from the service plane in addition to the evacuation paths and metrics calculated by the IoT cloud. DNs cannot handle complex tasks owing to their limited computational capabilities. Therefore, HREN does not allow overwriting centralized (global) decision by locally computed decisions. Accordingly, if a path sign is adjusted by the IoT cloud during the execution of HREN, this path sign cannot be adjusted later by its nearby DNs based on LEN procedure. This characteristic avoids recreating dead-end areas and leading evacuees to those areas. When changes are needed locally, DNs communicate with IoT cloud to get updates, if any. This guarantees to avoid any possible conflict between the distributed decisions calculated cooperatively by DNs and the global decisions that are informed by service plane and calculated remotely through IoT cloud in a centralized manner. It also allows integration with service plane and external rescuer when high personal risk is detected. Another interesting characteristic of this approach is that it only performs HREN on demand when high personal risk and dead-end points are detected, which eliminates the communication cost and delay encountered by centralized computation in normal situations.

V. EXPERIMENTAL EVALUTION AND ANALYSIS

This section presents the design and implementation of the simulation experiment. It discusses the different simulation scenarios, parameters, and performance factors.

In order to study and analyze the performance of the proposed approach, we implemented an event-driven simulator using MAT-LAB, presented in [19].

5.1. Simulation design and setup

In order to evaluate the performance of the proposed approach in terms of accuracy of making real-time decisions and adaption to current evacuation scenario, we consider different simulation scenarios. Moreover, a number of simulation variables are considered here in a way that mimics real-life problems, including the location of the hazard, the intensity of the hazard, number of evacuees, and the size of evacuation area.

The *Hazard location* has a substantial impact on the performance of emergency navigation algorithms. A well-designed evacuation approach can predict the path safety with respect to hazard location.

The *Hazard intensity* is very important for any evacuation approach in order to distinguish between different forms and intensities of hazards. The intensity (speed) of the spreading fire is constant through all the time [22]. In most cases, the incident itself (i.e. flames) and its side effects (smoke) may spread at different rates and different paths with complex correlations. For example, if a fire is the source of the hazard, the intensity is maximized because evacuees are affected by two forms of danger: flames and smoke. In our simulation, in order to assess the behavior of the proposed approach under different minor and major impact hazards, four different intensity values were considered to represent the intensity (rate of expansion) of the incident, including: 3, 5, 7, and 9. The intensity value 9 represents the highest intensity. It means the hazard expands 9 units of area (meters) within each unit of time (seconds) in all directions.

We design the first three experiments to determine the performance of the algorithms in small-scale, moderate-scale, and large-scale evacuation areas having 300 evacuees with respect to different hazard intensities. We design our fourth experiment to measure the performances of the three algorithms overcrowded areas with a different number of evacuees: 100, 300, and 500 evacuees. We design the fifth to measure performances of the algorithms when the number of exits gets minimized to 1. Finally, the sixth experiment studies the performances of the approach when the risk factor increases from normal condition to a certain increased level. In all these experiments, we describe results in terms of the following performance metrics:

(a) *The overall survival rate (or percentage of survivals)*: The percent-age of evacuees in the evacuation area who successfully evacuate through guided away from the hazard source to the exits.

(b) *Evacuation time:* The time duration taken to evacuate the people under the hazard from the hazardous area and locate them in safe zones.

Table 1 Percentages of survivals for the three approaches when evacuation area ranges from small to large area.

Evacuation area	Hazard intensity	Evacuation approach		
		LEN	HREN	DSPTD
Small area	3	97%	98%	93%
	5	96%	96%	85%
	7	95%	96%	83%
	9	96%	93%	87%
Medium area	3	99%	99%	96%
	5	99%	99%	96%
	7	99%	99%	92%
	9	98%	99%	89%
Large area	3	99%	100%	97%
	5	99%	100%	95%
	7	98%	94%	89%
	9	98%	92%	87%

(c) *Civilian casualties or fatalities* (number of dead civilians): We consider two different types of casualties: initial casualties and approach-resulted casualties. In our study, we are more concern about the approach-based casualties, which correspond to casualties caused by the evacuation approach not the initial causalities that are caused immediately by the hazard at the moment it occurred. This measure is calculated as the difference be-tween the total number of casualties at the end of the simulation and the initial casualties.

Based on the above described metrics, in the following subsections, we compare the performances of LEN, HREN, and DSPTD for randomly deployed evacuees and randomly deployed incidents in small, moderate, and large evacuation areas with different hazard intensities. At the end of this subsection, we highlight that there is a large number of evacuation and simulation variables that we can manipulate and measure in our simulation. However, in this study, we focus on a set of variables that relates to our study and considered interesting to evaluate and validate some aspects of our approach.

5.2. Simulation results for experiments 1, 2, and 3

This section presents results of the first three experiments for a total of 300 evacuees and different evacuation areas and hazard intensities. For the percentage of survivals, Table 1 provides a comparison of LEN, HREN, and DSPTD when the evacuation area ranges from small to large areas, and the hazard intensity ranges from 3 to 9.

Although DSPTD can always provide the shortest evacuation path for each individual, its performance is the worst, especially when the intensity of the hazard is medium-to-high for a small area of evacuation. This is because DSPTD tends to direct evacuees to the fastest route regardless the safety of that route. Therefore, evacuees may take the risk of navigating possible hazardous areas in order to reduce the evacuation time, and eventually suffers from hazards. Table 1 also shows that the proposed approach out-performs DSPTD in all considered areas. HREN and the LEN save almost all evacuees under different hazard intensities, except the evacuees under initial casualties (evacuees who are already located at the incident location, and therefore have no chance of evacuation).

Unlike DSPTD, LEN and HREN adapt to changes in the route as well as the environment, and they can dynamically redirect evacuees along paths away from the hazard. When the intensity was low to moderate, HREN achieves 100% survival rate while LEN achieves 99% survival rate. However, when the hazard in-tensity is very high, LEN mostly performs better than the HREN. This performance degradation is caused by the considered cloud

Table 2 Comparison of average evacuation time (in seconds) when the hazard intensity varies between 3 and 9, and evacuation area ranges from small area to large area.

Evacuation method	Intensity	Evacuation area= 100 × 100 m^2	Evacuation area = 200 × 200 m^2	Evacuation area = 300 × 300 m^2
LEN	Hazard intensity=3	8.0 s	16 s	22 s
HREN		7.4 s	14 s	21 s
DSPTD	Hazard	7.9 s	11 s	16 s
LEN	intensity=5	7.9 s	15 s	23 s
HREN		7.9 s	14 s	21 s
DSPTD	Hazard	6.0 s	11 s	16 s
LEN	intensity=7	7.9 s	15 s	23 s
HREN		7.6 s	15 s	24 s
DSPTD	Hazard	6 s	11 s	16 s
LEN	intensity=9	7.2 s	15 s	22 s
HREN		7.6 s	15 s	25 s
DSPTD		6 s	11 s	16 s

Note that DSPTD focuses on selecting the fastest paths to exits even if there are serious hazards near these paths. Besides, HREN requires a longer time under higher-intensity hazards. Consequently, it usually has longer evacuation time than LEN approach. This delay is a result of communication and computation delays caused by remotely executing HREN in the cloud in order to evacuate people who are located in safe dead-end areas.

Table 3 Comparison of average number of fatalities when the hazard intensity varies between 3 and 9, and evacuation area ranges from small area to large area.

Area	Intensity	LEN	HREN	DSPTD
Small area	3	7.9	6.9	21.7
	5	11.3	11.0	45.8
	7	13.7	13.2	49.8
	9	12.5	19.9	38.3
Medium area	3	3.6	1.9	13.2
	5	3.7	1.9	17.7
	7	3.9	1.9	24.9
	9	5.2	2.9	31.6
Large area	3	3	1	10
	5	3	1	15
	7	6	17	32
	9	6	24	39

communication delay and computation overhead encountered in making centralized decisions through the cloud, which delays evacuating evacuees located in dead-end areas when the hazard intensity is extremely high. Nevertheless, LEN and HREN outperform DSPTD. This happens due to our delicate blending of different pragmatic factors such as distances to exit(s) and incident, number of exit(s), risk, and hazard intensity Eqs. (1)–((3) in Section IV-B), which is not done in DSPTD.

Another important performance metric is the evacuation time. For a given evacuation area, DSPTD shows a fixed evacuation time regardless of the intensity of the hazard. This is due to the DSPTD's main aim of preserving a short evacuation time without considering path safety. In contrast, in our proposed algorithms, path safety is given the highest priority. In both LEN and HREN, evacuees are directed to the safest exit through the shortest path that has maximum safety compared to the alternative paths. Table 2 illustrate the average evacuation time for the different evacuation areas with hazard intensities of 3, 5, 7, and 9.

The ***number of fatalities*** is an important performance factor in evaluating the performance of any evacuation approach. Table 3 show the average number of fatalities for small, moderate-, and large evacuation areas and different hazard intensities. As shown in the figures, both LEN and HREN achieves substantially lower death rates compared to that of DSPTD. This behavior indicates that the performances of our proposed approach is better and, more importantly, stable in different evacuation areas. The results also show that when the hazard intensity is high and the area is large, DSPTD has high death rates compared to LEN and HREN.

In summary, LEN and HREN outperform DSPTD by 75% and 91%, respectively under low-intensity hazards in terms of the number of fatalities.

Table 4 Average percentage of survivals for the number of evacuees varies from 100 to 500.

Number of evacuees	LEN	HREN	DSPTD
100	96%	97%	91%
300	97%	98%	93%
500	97%	97%	94%

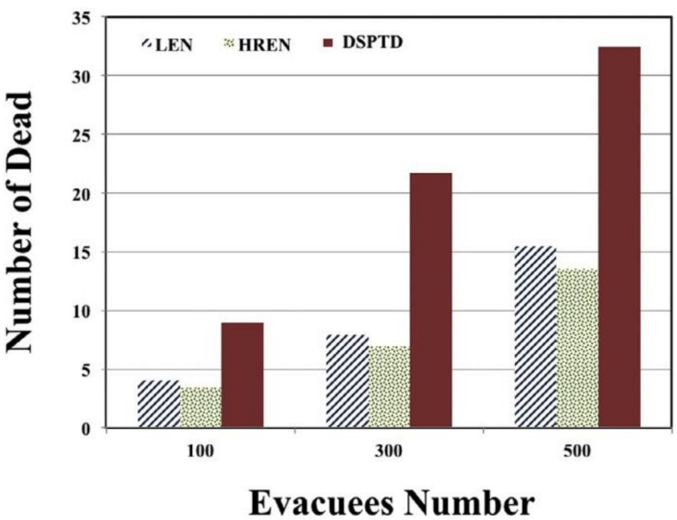

Fig. 4. Comparison of the number of fatalities when the number of evacuees between 100 and 500.

5.3. Simulation results for experiment 4

In this experiment, we evaluate the performance of LEN, HREN and DSPTD through varying the number of evacuees as 100, 300, and 500 while locating the hazard randomly with an intensity of 3. As presented in Table 4 and Fig. 4 results show that HREN and LEN exhibit substantially better performance in terms of survival rates and number of fatalities. Here, with a large number of evacuees, a longer time is required to navigate all evacuees to the exits, as shown in Fig. 5.

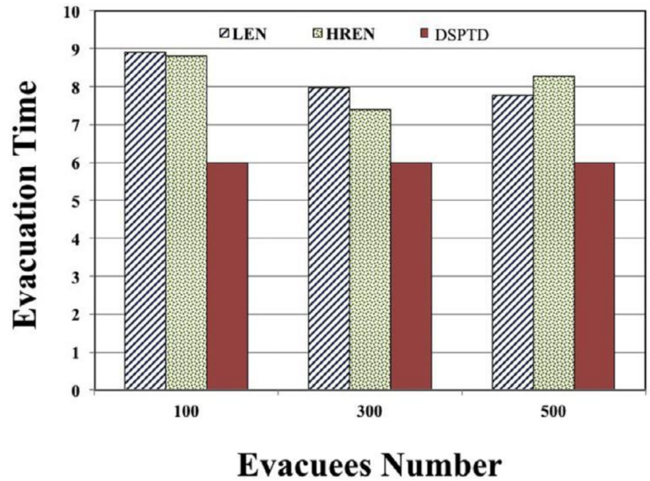

Fig. 5. Comparison the evacuation time when the number of evacuees between 100 and 500

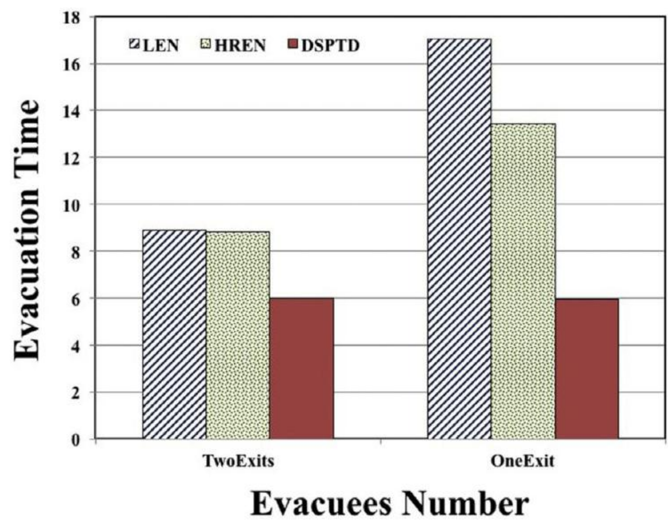

Fig. 7. Comparison of average evacuation time for different number of exits.

Table 5 Average percentages of average survival rates for different number of exits.

Exit availability	LEN	HREN	DSPTD
2 Exits	96%	97%	91%
1 Exit	79%	86%	81%

Table 6 Average percentage of survivals for different risk actors.

Risk factor	LEN	HREN	DSPTD
1	98.31%	99.68%	88.17%
2	98.73%	96.40%	91.99%
3	97.89%	90.64%	92.07%
4	97.89%	90.64%	92.07%

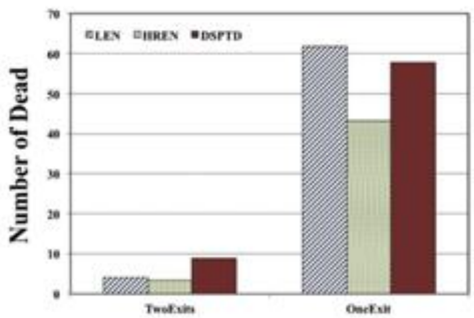

Fig. 6. Comparison of the number of fatalities for different number of exits.

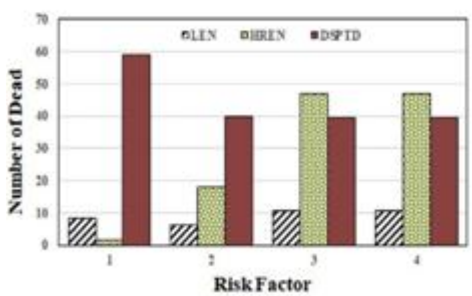

Fig. 8. Comparison of the number of fatalities for different risk factors.

5.4. Simulation results for experiment 5

This experiment studies the performance of the approach when only one exit in the building, or when one or more exits are blocked by a hazard. Here, we consider an evacuation area of 100 × 100m^2, having 300 evacuees. In such severe circumstances, the highest priority is to evacuate the civilians with the mini-mum death rate. Under these circumstances, simulation shows that HREN achieves the highest survival rate and, hence, the lowest death rate, as shown in Table 5 and Fig. 6. HREN achieves this performance improvement because it gives a higher priority for the safety of the evacuation path over that of the speed of evacuation.

The DSPTD experiences the highest death rate due to its priority assignment to the evacuation time over the safety of the path. Accordingly, similar to the earlier cases, our proposed approach re-quires a bit increased evacuation time, as shown in Fig. 7.

5.5. Simulation results for experiment 6

This experiment studies the performances of the approach when the risk factor increases from normal condition to a certain increased level (for example, 4 in our case). Here, we consider an evacuation area of 300 × 300m^2, having 500 evacuees, 2 exit points, and the hazard intensity to be 7.

Under these circumstances, the simulation results show that LEN achieves the highest survival rate and, hence, the lowest death rate, as shown in Table 6 and Fig. 8. Here, LEN achieves the best performance in a stable manner for most of the risk factors under analysis.

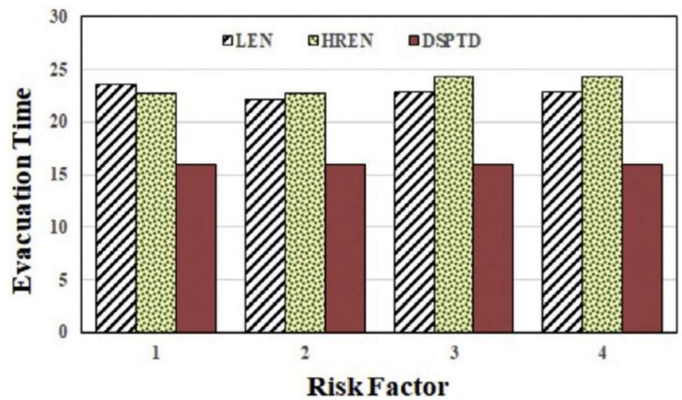

Fig. 9. Comparison of evacuation time for different risk factors.

Nonetheless, similar to the earlier cases, our proposed approach requires a bit increased evacuation time com-pared to DSPTD as shown in Fig. 9.

5.6. Summary of simulation results

To conclude, in comparison with the DSPTD, our proposed approach achieves overall higher survival rates as a result of its ability to tailor paths to evacuees with respect to the safety of the path leading them farthest from the hazard. It is worth mentioning that our proposed approach takes a slightly longer time for evacuation compared to DSPTD, as we consider the safety of the evacuation paths in addition to the speed of the evacuation process. Thus, our proposed approach might guide an evacuee through longer paths to avoid zones at higher risk of hazard. This happens as the main aim of our proposed approach is to find the best (safest) path available, not the fastest path, as done by DSPTD. Therefore, DSPTD experiences higher death rates in spite of having faster evacuation mainly because it does not adapt to real-time changes to the hazard locations and always directs evacuees to the nearest exit by searching for the fastest path with no prior hazard calculation.

VI. FEASIBILITY AND EFFECTIVENESS OF PROPOSED SOLUTION

In recent years, many solutions using wireless sensor networks and cloud computing based fire management systems have been proposed and some of them are being used in a feasible manner in the industries [26–31]. Here, only a few of them focus on determining which evacuation route is the best only from the perspective of evacuation time. Thus, these solutions ignore an important aspect of safety of the suggested evacuation path, which we focus in this paper. Other comparable solutions exploit RFID and mobile for the navigation of the evacuees [28,29]. In these solutions, the evacuees must have a mobile application installed in their mobile devices. Therefore, in these cases, the evacuees may get stuck because of low computation power in the local mobile devices (where the integration of cloud computing comes into play as presented in this paper). Besides, for the other existing solutions, they are limited to only monitoring fire hazards [26,33] or only for the rescuing task [32] having no focus on suggesting evacuation paths to evacuees.

VII. CONCLUSION

This paper proposes real-time routing algorithms to increase the survival rate of an emergency evacuation process. Here, we employ a hybrid solution combining IoT and cloud computing to perform navigation to predict safe dead-end problems. The IoT-based approach performs the task of suggesting safe evacuation paths in an emergency scenario that can be handled with the computation power of IoT. In case the scenario goes beyond a certain complexity demanding high computational power, we propose to enable a cloud-based approach to solve this problem remotely in a centralized manner. Examples of such happening of delegation to the cloud include cases when the evacuees reach in dead ends.

We perform performance evaluation of our proposed approach through simulation. Here, we adopt a fire model to predict the hazard spread. Our calculation of safe routes in the simulation is based on the initial distribution of evacuees, their distances from the hazard, their distances to the exit, and the intensity of the hazard.

VIII. ACKNOWLEDGMENTS

It is with the greatest pride that we publish this paper. At this moment, it would be unfair to neglect all those who helped me in the successful completion of this paper. I would also like to thank all the faculties who have cleared all the major concepts that were involved in the understanding of techniques behind my paper.

REFERENCES

[1] O.J. Akinwande, H. Bi, E. Gelenbe, Managing crowds in hazards with dynamic grouping, IEEE Access 3 (2015) 1060–1070.

[2] F. Wu, C. Rudiger, M.R. Yuce, Design and field test of an autonomous IoT WSN platform for environmental monitoring, in: Presented at ITNAC Conference, Melbourne, VIC, 2017, pp. 1–6.

[3] H. Bi, E. Gelenbe, Routing diverse evacuees with cognitive packets, in: 2014 IEEE International Conference on Pervasive Computing and Communication Workshops (PERCOM WORKSHOPS), Bu A dapest, 2014, pp. 291–296.

[4] G. Barrenetxea, F. Ingelrest, G. Schaefer, M. Vetterli, Wireless sensor networks for environmental monitoring: the sensorscope experience, in: 2008 IEEE In-ternational Zurich Seminar on Communications, Zurich, 2008, pp. 98–101.

[5] W. Wang, K. Lee, D. Murray, Integrating sensors with the cloud using dynamic proxies, in: 2012 IEEE 23rd International Symposium on Personal, Indoor and Mobile Radio Communications - (PIMRC), Sydney, NSW, 2012, pp. 1466–1471.

[6] B. Perumal, P. Rajasekaran, H.M. Ramalingam, WSN integrated cloud for auto-mated telemedicine (ATM) based e-healthcare applications, 4th International Conference on Bioinformatics and Biomedical Technology, Feb. 2012.

[7] R. Fantacci, T. Pecorella, R. Viti, C. Carlini, A network architecture solution for efficient IoT WSN backhauling: challenges and opportunities, IEEE Wirel. Com-mun. 21 (4) (2014) 113–119.

[8] M. Amarlingam, Pradeep Kumar Mishra, KVV Durga Prasad, P. Rajalakshmi, Compressed sensing for different sensors: a real scenario for WSN and IoT, in: Internet of Things (WF-IoT), 2016 IEEE 3rd World Forum on, 2016,289–294.

[9] Anonymous. “Benefits of cloud computing to wireless sensor networks,” ISSN 2001-6298.

[10] K. Ahmed, M. Gregory, Integrating wireless sensor networks with cloud com-puting, in: Seventh International Conference on Mobile Ad-hoc and Sensor

Networks, 2011, pp. 364–366.

[11] M. Qiu, Z. Ming, J. Wang, L.T. Yang, Y. Xiang, Enabling cloud computing in emergency management systems, IEEE Cloud Comput. 1 (4) (2014) 60–67.

[12] P. Boguslawski, L. Mahdjoubi, V. Zverovich, F. Fadli, A dynamic approach for evacuees' distribution and optimal routing in hazardous environments, Autom. Construct. 94 (2018) 11–21.

[13] T. Hara, M. Sasabe, S. Kasahara, Geographical risk analysis based path selection for automatic, speedy, and reliable evacuation guiding using evacuees' mobile devices, J. Ambient Intell. Human. Comput. (2018) 1–10.

[14] J.C. Chu, A.Y. Chen, Y.F. Lin, Variable guidance for pedestrian evacuation con-sidering congestion, hazard, and compliance behavior, Transport. Res. Part C 85 (2017) 664–683.

[15] L. Karagiannidis, F. Misichroni, Y. Damigos, A. Tsertou, A. Amditis, A novel and interoperable communication gateway implementation for evacuation systems, in: 2016 International Wireless Communications and Mobile Computing Con-ference (IWCMC), Paphos, 2016, pp. 1045–1050.

[16] A. Depari, A. Flammini, D. Fogli, P. Magrino, Indoor localization for evacuation management in emergency scenarios, in: 2018 Workshop on Metrology for In-dustry 4.0 and IoT, Brescia, 2018, pp. 146–150.

[17] Hong L., B. Xu, D. Lu, G. Zhang, A path planning approach for crowd evacua-tion in buildings based on improved artificial bee colony algorithm, Appl. Soft Comput. 68 (2018) 360–376.

[18] H. Bi, E. Gelenbe, A cooperative emergency navigation framework using mobile cloud computing, in: Information Science and Systems, Springer International Publishing, 2014, pp. 41–48.

[19] N. Al-Nabhan, N. Al-Aboody, H. Rawishidy, Adaptive wireless sensor network and cloud-based approaches for emergency navigation, in: Proc. IEEE LCN 2017, Singapore, 2017.

[20] H. Bi, E. Gelenbe, Cloud-enabled emergency navigation using faster-than-real– time simulation, in: PerCom Workshops, 2015, pp. 475–480.

[21] L.-W. Chen, J.-J. Chung, Mobility-aware and congestion-relieved dedicated path planning for group-based emergency guiding based on internet of things tech-nologies, IEEE Trans. Intell. Transport. Syst. 18 (9) (2017) 2453–2466.

[22] X. Dai, L. Jiang, J. Maclean, S. Welch, A. Usmani, A conceptual framework for a design travelling fire for large compartments with fire resistant islands, in: The 14th International Interflam Conference (Interflam 2016), London, 2016,1039–1050.

[23] Tasnim and I. Nawar, "Fire in garments factory of Bangladesh", 2018. [On-line]. [Accessed: 10- Mar- 2019]. Available: https://www.adpc.net/igo/contents/iPrepare/iprepare-news_id.asp?ipid=220.