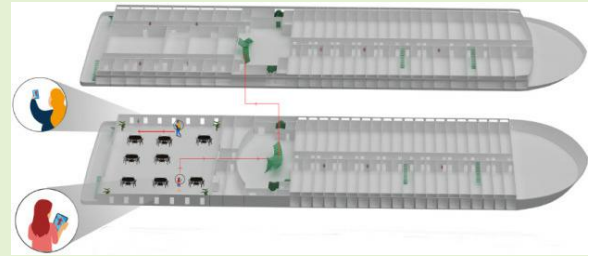


WEND: An Efficient WSN-assisted Emergency Navigation Algorithm for Dynamic Hazardous Ship Indoor Environments

Yuting Ma, Kezhong Liu, *Member, IEEE*, Mozi Chen, *Member, IEEE*, Kehao Wang, Kai Zheng, Xuming Zeng

Abstract—A reliable and efficient emergency evacuation of a damaged cruise ship is essential for passenger safety. Although many advanced evacuation approaches based on wireless sensor networks (WSNs) are capable of exploring dynamic environmental hazards and provide real-time navigation service, such approaches are mainly used in buildings on land, taking no account of the complex ship structure, deadline for ship survival, and dynamic ship inclination, and therefore may fail to evacuate passengers before ship capsizing. This paper proposes WEND, an efficient WSN-assisted emergency navigation algorithm for dynamic hazardous ship indoor environment, which informs each passenger about a hazard-avoid route that minimizes the total dynamic typical delay with error bound guarantee while meeting the deadline for ship capsizing in a real-time manner. To achieve this aim, WEND investigates ship interior layout to construct a 3D topological model and analyzes the time-history of the roll motion of a damaged passenger ship to evaluate ship survival time and the dynamics of pedestrian movement. Then, an efficient adaptive emergency navigation algorithm building on the ideas of Hassin's algorithm is presented to provide a hazard-avoid path from each passenger's current location to boarding stations such that the worst-case delay along this path is no greater than the specified deadline, and the total dynamic typical delay is small. We evaluate our algorithm by conducting extensive simulations. The results demonstrate that WEND improves the navigation success ratio by approximately 30% and 10% compared with the state-of-the-art navigation methods, namely the Expected Number of Oscillations (ENO)-based oscillation-free method (OPEN) and the deadline-aware adaptive emergency navigation strategy (ANT), respectively.



Index Terms—Dynamic navigation model, Approximation algorithm, Ship evacuation, Wireless sensor networks.

I. INTRODUCTION

EMERGENCY evacuation of a passenger ship becomes very important, especially considering the enormous loss of life in case of passenger ship disasters [1], [2]. Costa Concordia disaster left 32 passengers and crew dead and more than 4,000 traumatized [3]. The sinking of the Al-Salam Boccaccio 98 ferry took the lives of more than 1,000 people [4]. According to a safety technical investigation report submitted by the Italian Ministry of Infrastructure and Trans-

port, inefficient emergency evacuation is considered one of the primary factors for such serious casualties [5]. Therefore, an efficient and safe ship evacuation strategy plays a critical role in protecting passengers' lives when ships encounter accidents. Ship evacuation is very complex because of the ship's complicated internal structure, the dynamic evacuation environment, and the limited ship survival time.

Most of the existing ship evacuation research focuses on developing simulation programs that can handle complex navigation scenarios, i.e., the influence of ship motion and human behaviors [6]–[9]. However, the simulation-based approaches are high-priced and time-consuming, which has a negative performance impact on the time-critical ship evacuation service. In addition, the results from these approaches are only effective for the previously specified damage case. That is to say, they cannot provide appropriate navigation service for evacuees in a real-time manner, considering dynamic emergency scenarios.

Wireless Sensor Networks (WSNs), which can be widely deployed in the fields of interest, are capable of automatically exploring and interacting with the dynamic environment. Therefore, we can incorporate WSNs into emergency navigation systems to extensively monitor the ever-changing

Manuscript is submitted on May 2, 2022. This work was supported in part by the National Natural Science Foundation of China (NSFC) under Grant 51979216, and partially by the Natural Science Foundation of Hubei Province, China, under Grant 2021CFA001 under Grant BS123456

K. Liu is with the School of Navigation, Wuhan University of Technology, Wuhan 430063, China; the National Engineering Research Center for Water Transport Safety, Wuhan 430063, China; and the Hubei Key Laboratory of Inland Shipping Technology, Wuhan 430063, China (e-mail: kzliu@whut.edu.cn).

Y. Ma, M. Chen, K. Zheng, and X. Zeng are with the School of Navigation, Wuhan University of Technology, Wuhan 430063, China.

K. Wang is with the School of Information Engineering, Wuhan University of Technology, Wuhan 430063, China.

*Corresponding author: Mozi Chen (chenmz@whut.edu.cn).

ship indoor environment and then provide real-time navigation service to users equipped with portable devices such as Personal Digital Assistants (PDAs) and smartphones [10]–[16]. However, considering the complex 3D model of ship internal structure, the dynamic ship inclination, and the deadline for ship sinking, most existing WSN-assisted emergency navigation approaches for land cannot be directly applied to ship evacuation.

According to presently valid regulations passenger ships need sufficient hydrostatic stability to survive from certain damage cases (e.g., fire, collision, and grounding) [17]–[19]. However, the required damage stability does not, in all cases, guarantee the survival of a ship over an infinite period of time, especially if an accident takes place at unfavorable sea states or weather conditions. In such a case, the time available for ship evacuation is limited. That is, passengers and crew must escape from their current locations to a specified boarding point for lifeboats or Marine Evacuation Systems within a specified deadline. Otherwise, there is very little chance of surviving for them. However, to the best of our knowledge, the existing WSN-assisted navigation methods do not take the hard deadline into account, which can dramatically jeopardize users' chances of survival. Specifically, these approaches consider the impact of variations of hazards and aim to guide users to escape without oscillations while away from hazards, but they ignore navigation efficiency, leading to passengers' remaining in danger for a long period of time and thus missing the deadline for ship survival [14].

For general land-based buildings, the traversal speed of evacuees is only affected by their own properties (e.g., their age, gender, physical and psychological status) [20], [21] and the density of passageways which is related to the distribution and spatial-temporal movement of evacuees [22]. While on a passenger ship, there is a special feature for the movement speed of passengers distinguishing it from that in land-based buildings since the fact that evacuees escape on an unstable surface. That is, the walking speed on a passenger ship is not only influenced by the above-mentioned factors, but also severely influenced by the changing slope angles, which is unrelated to the passengers' personal attributes or their spatiotemporal movement [23], [24]. When the inclination angle varies over time, passengers tend to feel dizzy, and their gait alters as well, all of which results in a change in individual walking speed. We can evaluate the dynamic passengers' movement during evacuation using simulated data from the following organizations: the Korea Research Institute of Ship and Ocean Engineering, the Netherlands Organisation for Applied Scientific Research (TNO), the Swiss Federal Institute of Technology, and the Monash University in Australia [25], [26]. However, the existing WSN-assisted navigation systems do not fully consider the dynamics of passenger walking speeds due to changing inclination states, which can result in the variation of transit time on the same passageway. They utilize the path length, which is considered to maintain invariable in time, to design different metrics of path planning (e.g., the shortest route, the minimum exposure route, and the oscillation-free route to navigate evacuees), and thus the paths may be time-consuming or impassable in practice [14], [27]–[29].

In addition, most of the existing WSN-based navigation approaches deal with only 2D sensing fields and take no account of actual arrangement of navigation environment [30], [31]. These approaches explore physical obstacles in the environment and construct the route graph during run-time phase, which spends a significant amount of time due to the extremely complicated internal structure of a cruise ship, and thus cannot offer real-time navigation services for trapped users.

In this paper, we propose WEND, an efficient WSN-assisted emergency navigation algorithm for dynamic hazardous ship indoor environments, which can, in real-time or near real-time, provide passengers with hazard-avoid navigation paths that achieve no more than $(1+\epsilon)d_T^*(\mathcal{T})$ ¹ typical delay while simultaneously guaranteeing to respect the specified deadline even if the worst-case delay is encountered in traversing the path. To achieve this purpose, WEND first constructs the 3D topological model of a real-world passenger ship and then deploys sensors based on the 3D model to monitor dynamic hazards and serve as landmarks to guide passengers to a destination. Then WEND analyzes the effect of dynamic ship inclination on delay and combines the impact of crowd density, flood water depth, as well as locations of segments and damages on a passenger ship to determine the typical delay and the worst-case delay bound across each path accordingly. Based on the two delay parameters and the monitored hazard situation and movement, an adaptive emergency navigation strategy with low time complexity, motivated by a fully polynomial approximation scheme, namely, Hassin's Algorithm [32], [33], is proposed. We evaluate WEND by conducting extensive simulations. The results demonstrate that WEND improves the navigation success ratio by approximately 30% and 10% compared with the state-of-the-art navigation methods, namely, the Expected Number of Oscillations (ENO)-based oscillation-free method (OPEN) and the deadline-aware adaptive emergency navigation strategy (ANT), respectively.

The main contributions of our study are summarized as follows:

- To the best of our knowledge, this is the first WSN-assisted navigation strategy taking 3D topological structure of a real passenger ship into consideration. We utilize a Voronoi Diagram (VD)-based method to automatically tessellate complex rooms (i.e., rooms with more than one door in the ship), and construct a variable navigable network before the application of our path finding algorithm.
- We synthetically consider the dynamics of both hazards and changing walking environments and construct a dynamic navigation model.
- Given the serious threat of ship capsizing to passengers' survival, we design WEND, an efficient emergency navigation algorithm that can, in near real-time, find hazard-avoid paths with no more than $(1+\epsilon)d_T^*(\mathcal{T})$ total typical delay while guaranteeing to respect the deadline for ship capsizing under all circumstances.
- We theoretically analyze the upper and lower bounds on

¹ ϵ is an approximation factor, $p^*(\mathcal{T})$ is the optimal path defined in Section III-B, $d_T^*(\mathcal{T})$ denotes the typical delay of path $p^*(\mathcal{T})$

the typical delay of the path provided by our approximate algorithm and evaluate WEND using extensive simulations. Experimental results demonstrate that WEND outperforms the state-of-the-art evacuation methods in terms of navigation efficiency and navigation success ratio.

The remainder of the paper is organized as follows. Section II presents our motivation and preliminary. Section III presents the dynamic navigation model and problem formulation. The detail of the design of our method is introduced in Section IV. Section V presents theoretical analyses and proofs on several important issues. We evaluate the performance of our approach through extensive simulations in Section VI. Finally, Section VII concludes this work.

II. MOTIVATION

In this section, we motivate our design by showing the three particular characteristics of ship evacuation, which are not considered in prior works.

A. Deadline-aware emergency navigation

The main difference of emergency evacuation between ships and other scenarios is that there exist specific deadlines for the ship tasks, i.e. the ship sinks or is too banked to move. Our prior work, i.e., Deadline-aware Adaptive Emergency Navigation Strategy for Ship Evacuation with Wireless Sensor Networks (ANT), takes into account the two features of ship passenger evacuation and assumes a model that characterizes each passageway by a worst-case delay (i.e., a guaranteed upper bound on the delay) and a typical-case delay (i.e., an estimate of the delay that is typically encountered upon traveling a passageway) [34]. The performance objective considered in ANT is to identify the next node that minimizes the typical delay while respecting the deadline for ship capsizing and avoiding hazards under all circumstances. ANT can select different directions for different passengers based on their actual walking delay, which will be affected by dynamic ship motion. However, ANT does not consider the variation of the estimate of the typical delay of each segment. That is, the look-up tables in ANT are established only using the initial typical delays, and thus escaping along the provided guiding direction may not be optimal. If we only depend on the periodic recalculation of look-up tables, whose construction is extremely time-consuming, there will be a negative performance impact on the time-critical ship evacuation service. In addition, ANT deals with only 2D sensing fields and takes no account of the actual arrangement of the navigation environment.

B. Effect of dynamic ship inclination on walking time

Due to the ship's complex indoor environment, the physical distance may not represent the closeness between two landmarks corresponding to two sensor nodes. Therefore, in our design, we measure the closeness in terms of passenger walking time that, in some cases (see Fig. 1), is not proportional to the physical distance. As mentioned above, there is a deadline for ship evacuation by which passengers

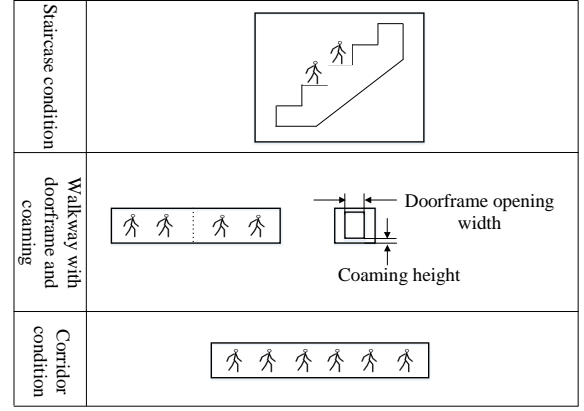


Fig. 1: The examples of walking condition on a passenger ship

need to escape from their current locations to the specified boarding point. Otherwise, it denotes navigation failure for them. Therefore, except for the estimate of the delay typically encountered across a link between two nodes, each link is also characterized by an estimate of the upper bound on maximum delay on it, which is trustworthy at very high assurance levels. The calculation procedure of the two delay parameters is introduced in detail in Section IV-C.

When considering the effect of dynamic ship inclination, it becomes even more challenging to design navigation routes. Because of the inflow of water through any opening produced by damage (e.g., a collision), the mass, the center of gravity, and the moment of inertia of the ship will vary accordingly, which causes the changing slope angle and thus alters the typical delay on each link. Therefore, our algorithm needs to handle the scenario where typical delays are dynamic.

In addition, according to the investigation by TNO, the decrease in walking speed for dynamic motion conditions is less than for the static list. [35] For this reason, the effect of the ship pitching and rolling on walking time is not taken into account in our paper. Specifically, typical delays on certain links are represented as a monotonically decreasing staircase function whose staircase height and width are determined by the specific evacuation scenario.

C. Effect of complex 3D ship indoor environment

When facing the rapid increase in ship size and the more complicated topology of ship indoor environment, it becomes more challenging to provide real-time navigation suggestions to passengers. Large cruise and passenger ships, typically with 6 to 16 stories and having the capacity to accommodate thousands of passengers (e.g., Harmony of the Seas, a Royal Caribbean cruise ship, which can contain 6,360 passengers), are packed with restaurants, entertainment venues, and accommodation categories [36]. Exploring the internal topological structure of such ships and then constructing a route graph during the run-time phase is extremely time-consuming and may cause heavy traffic overhead. Therefore, the provision of situational awareness in the pre-processing phase for emergency evacuation becomes particularly important. That is, we

need to extract the 3D topological model of a passenger ship before the run-time phase to find all feasible navigation routes containing corridors, staircases, and elevators.

In addition, due to the large-scale network configurations, exact navigation algorithms will be computationally expensive and time-consuming. Therefore, in our paper, we need to design an efficient approximation scheme for ship emergency navigation.

III. MODEL & PROBLEM FORMULATION

In this section, we first introduce the navigation model and then formulate the problem.

A. Model and definitions

In this part, we provide an in-depth description of the navigation model, with notations presented in Table 1.

TABLE I: Notations and definitions used in the model.

Notation	Meaning
\mathcal{V}	Set of landmarks
\mathcal{E}	Set of segments
\mathcal{V}_h	Set of hazardous landmarks
\mathcal{V}_g	Set of general landmarks
v_u	User landmark
v_e	Exit landmark
$\overrightarrow{v_i v_j}$	Segment between landmark v_i and v_j
$d_T^{\overrightarrow{v_i v_j}}(t)$	Typical delay of a segment at time point t
$d_W^{\overrightarrow{v_i v_j}}$	Worst-case delay of a segment
$\mathcal{D}_h^v(t)$	Set of the shortest interval of hazardous arrival time at segments at time t
$D_h^{\overrightarrow{v_i v_j}}(t)$	The shortest interval of hazardous arrival time at a segment $\overrightarrow{v_i v_j}$ at time t
\mathcal{E}_v	Set of neighbor segments of a landmark
p_{ue}	Navigation path from user landmark v_u to exit landmark v_e
$d_T^p(\mathcal{T})$	Typical delay of p during time interval \mathcal{T}
d_W^p	Worst-case delay of p

In order to provide real-time navigation service, a WSN is deployed in the navigation scenario according to the 3D topological model of ship indoor environment, which is derived in the pre-processing phase. The WSN is modeled by a directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where \mathcal{V} is the set of landmarks and \mathcal{E} denotes the set of walking paths between two landmarks (called segments in our design). Each landmark v ($v \in \mathcal{V}$) corresponds to a sensor node. There are two types of landmark roles: hazardous landmark \mathcal{V}_h and general landmark \mathcal{V}_g , which indicate a landmark inside or outside of hazardous regions. The hazardous region is modeled as a convex hull of the subset of hazardous landmarks. We assume that there are a finite number of hazardous regions threatening users' safety, and regarding the pattern of dynamics of hazard, we suppose that hazardous regions can only expand as time passes. Note that the landmark's role may transform with the dynamics of hazards. The landmark that a user currently arrives at is defined as the user landmark ($v_u \in \mathcal{V}$), and the landmark closest to a muster station is considered as the exit landmark ($v_e \in \mathcal{V}$). Fig. 2 shows the scenarios of emergency navigation using a WSN at different times and corresponding navigation models.

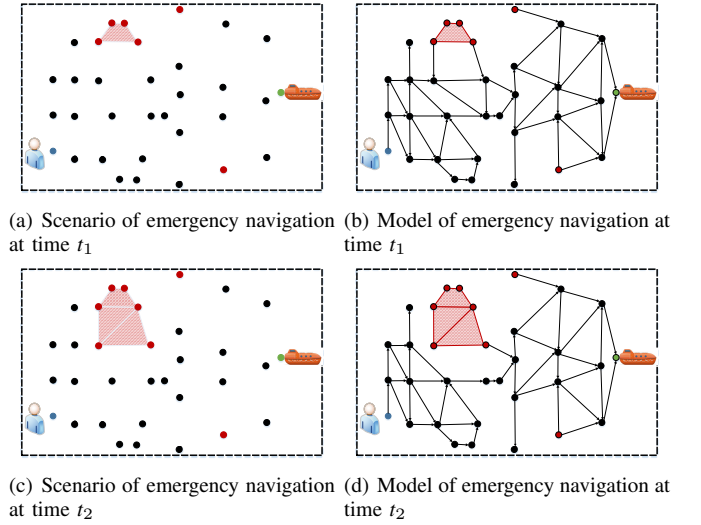


Fig. 2: Scenarios of emergency navigation and corresponding navigation models. The black circles denote general landmarks, the red circles represent hazardous landmarks, the green circle indicates the exit landmark, and the blue landmark represents the user landmark.

In our scenarios, there is an exit that users are required to lead to.

Each directed segment $\overrightarrow{v_i v_j}$ (i.e., the directed walking path between a landmark v_i and another landmark v_j) is characterized by two delay parameters: typical delay $d_T^{\overrightarrow{v_i v_j}}(t)$ ($d_T^{\overrightarrow{v_i v_j}}(t) > 0$) and worst-case delay $d_W^{\overrightarrow{v_i v_j}}$ ($d_W^{\overrightarrow{v_i v_j}} \geq d_T^{\overrightarrow{v_i v_j}}(t)$), which represent an estimate of the delay that is typically encountered, and a guaranteed upper bound on the maximum delay, across the segment $\overrightarrow{v_i v_j}$, respectively. Considering the effect of the continuously increasing heel angle of a damaged passenger ship on pedestrian movement during navigation process, the typical delay on each segment will vary dynamically. Each landmark v is presented as a three-tuple: $\langle \text{ID}, \text{LC}, \mathcal{D}_h^v(t) \rangle$. ID is a unique landmark identifier, which is assigned when a sensor network is deployed, and LC denotes the location coordinate of a landmark, which is assumed to be available in our navigation scenario. $D_h^{\overrightarrow{v_i v_j}}(t)$ represents the shortest interval of time during which hazard expands from the initial location until encroaching on the neighbor segment $\overrightarrow{v_i v_j}$ of the landmark v_j . $\mathcal{D}_h^v(t)$ is defined as follows:

$$\mathcal{D}_h^v(t) \triangleq \left\{ D_h^{\overrightarrow{v_i v_j}}(t) \mid \overrightarrow{v_i v_j} \in \mathcal{E}_{v_j} \right\}$$

\mathcal{E}_{v_j} represents the set of the neighbor segments of the landmark v_j , which is defined as follows:

$$\mathcal{E}_{v_j} \triangleq \{ \overrightarrow{v_i v_j} \mid \overrightarrow{v_i v_j} \in \mathcal{E} \}$$

The calculation procedure of $D_h^v(t)$ is introduced in detail in Section IV-C. For segment $\overrightarrow{v_i v_j}$ ($\overrightarrow{v_i v_j} \in \mathcal{E}_{v_j}$, $v_j \neq v_e$), if $D_h^{\overrightarrow{v_i v_j}}(t) \neq +\infty$, then hazard is approaching it. Otherwise, it means hazards do not influence the segment.

A navigation path p_{ue} consists of a landmark sequence, on which the user landmark v_u is the header and the exit landmark v_e is the rear:

$$p_{ue}(\mathcal{T}) \triangleq \langle v_u, \dots, v_i, v_j, \dots, v_e \rangle$$

where $p_{ue}(\mathcal{T})$ denotes the landmark sequence which is dynamically updated according to the dynamics of typical delay and stored in every landmark between a user landmark and an exit landmark. \mathcal{T} indicates the cumulative time which is the sum of Δt of the segments on $p_{ue}(\mathcal{T})$. The typical delay and the worst-case delay of a path $p(\mathcal{T})$ are defined as follows, respectively:

$$d_T^p(\mathcal{T}) \triangleq \sum_{\forall \overrightarrow{v_i v_j} \in p(\mathcal{T})} d_T^{\overrightarrow{v_i v_j}}(t)$$

and

$$d_W^p(\mathcal{T}) \triangleq \sum_{\forall \overrightarrow{v_i v_j} \in p(\mathcal{T})} d_W^{\overrightarrow{v_i v_j}}$$

B. Problem formulation

The objective of our emergency navigation strategy is to plan a path $p^*(\mathcal{T})$ apart from hazardous regions, through which a user can arrive at the exit landmark within a specified deadline under all circumstances, while experiencing the minimum typical delay from initial user landmark to exit landmark. Below, we formulate the properties we desire in our navigation strategy:

- **Path safety guaranteed.** Let \mathcal{P}_{je} denote the set of paths from v_j to v_e and \mathcal{V}_i^s be the set of safe neighbor landmarks of v_i . \mathcal{V}_i^s is defined as follows: if $\exists p_{je} \in \mathcal{P}_{je}$, $\forall v_n \in p_{je}$ ($v_n \neq v_j$) satisfies:

$$d_W^{\overrightarrow{v_i v_j}} + d_W^{p_{je}} \leq D_h^{\overrightarrow{v_m v_n}}(t), v_m \in p_{je} \wedge \overrightarrow{v_m v_n} \in \mathcal{E}_{v_n}, \quad (1)$$

then v_j is regarded as a safe neighbor landmark of v_i , and all safe neighbors of v_i form the set \mathcal{V}_i^s . Let \mathcal{P}_{ie}^s be the set of safe paths from v_i to v_e . We have the following expression for \mathcal{P}_{ie}^s : with respect to a path p_{ie} , if and only if for each landmark v_i that lies on p_{ie} , $v_j \in \mathcal{V}_i^s$, then p_{ie} is considered as a safe path from v_i to v_e , where $v_j \in p_{ie}$ and $\overrightarrow{v_i v_j} \in \mathcal{E}_{v_j}$, and all safe paths from v_i to v_e form the set \mathcal{P}_{ie}^s .

- **Path efficiency guaranteed.** Let $d_T^{*p_{ie}}(t_i)$ be the minimum typical delay that can be achieved from v_i to v_e while simultaneously guaranteeing the avoidance of both dynamic hazards and exceeding a hard deadline with regard to time t_i (timestamp of arriving at the landmark v_i). $d_T^{*p_{ie}}(t_i)$ is defined as follows:

$$d_T^{*p_{ie}}(t_i) \triangleq \min \left\{ d_T^{\overrightarrow{v_i v_j}}(t) + d_T^{*p_{je}}(t_j) \right\}$$

where $v_j \in \mathcal{V}_i^s$. That is, $d_T^{*p_{ie}}(t_i)$ is the minimum value of the typical delay from v_i to v_j plus the minimum typical delay that can be achieved from v_j to v_e while guaranteeing the avoidance of both dynamic hazards and exceeding a hard deadline with regard to time t_j .

Then we define π_i as the optimal neighboring landmark of v_i which determines the value of $d_T^{*p_{ie}}(t)$ ($\overrightarrow{v_i \pi_i} \in \mathcal{E}$), and thus the safe path $p_{ue} \in \mathcal{P}_{ue}^s$ can be called the optimal path $p^*(\mathcal{T})$, if and only if for each landmark v_i that lies on p_{ue} , $v_j = \pi_i$.

- **Algorithm efficiency guaranteed.** An emergency navigation scheme demands a time-critical response with respect to the provision of guidance. The extensive time cost required to run the navigation strategy has a negative impact on ship evacuation service and thus is undesired.

IV. WEND DESIGN

In this section, we propose an efficient approximation scheme, WEND, for emergency navigation in ship indoor environments, which fulfills all the goals in Section III-B. We first present an overview of WEND. The second part introduces the 3D navigation network construction of a passenger ship. The third part describes the calculation procedure of the parameters of our dynamic navigation model. Finally, we present the process of path generation and navigation, followed by an example of our navigation scheme.

A. Design overview

Fig. 3 shows an architecture overview of the proposed navigation scheme. There are two modules in our scheme: parameter predictor and routing predictor. These modules are integrated on a path planning server, supporting a navigation workflow consisting of several phases, namely initialization, parameter estimation, and path generation.

We considering the following: a trapped passenger equipped with a portable device is navigated towards a muster station on a ship. The portable device uses a radio frequency (RF) module to access the WSN deployed in advance. First, the parameter predictor evaluates the typical delay $d_T^v(t)$ and worst-case delay $d_W^{\overrightarrow{v_i v_j}}$ on segments, and $D_h^v(t)$, the set of the shortest interval of time during which hazards encroach on segments. Based on the dynamics of both emergency and ship inclination, $d_T^{\overrightarrow{v_i v_j}}(t)$ and $D_h^v(t)$ can be updated in a real-time pattern. Second, the routing predictor is triggered to calculate the path $p^*(\mathcal{T})$ for each user according to our proposed path planning algorithm. The trapped user can avoid hazardous regions and reach the exit within a specified deadline under all circumstances while experiencing a no more than $(1+\epsilon)d_T^*(\mathcal{T})$ typical delay from the initial user landmark to exit landmark in accordance with the indicators on the device.

B. 3D navigation network construction.

In this part, we present the detail of generating a navigation network in a 3D ship indoor environment, which is the first step of WEND. This network includes only connection available for pedestrian movement within a ship. That is to say, adjacent floors are connected only by vertical connection representing staircases, and spatial relationships between adjacent rooms sharing a wall with a door are reflected in the network.

If the navigation network in simple models, e.g., when it consists of ‘cubical’ rooms with one door or two doors located on opposite walls, is generated based on the Poincaré duality [37]. In the network, a room is represented by a dual node (corresponding to a landmark), a wall with a door is a dual

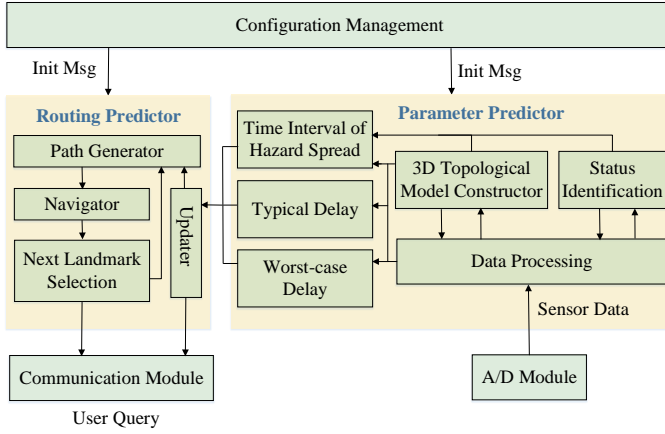


Fig. 3: Overview of our navigation architecture

edge (corresponding to a segment) bounded by dual nodes representing adjacent rooms.

However, the above network cannot be effectively used for navigation routes when rooms have a complex shape and many doors. A method of navigation network construction in complex rooms, based on the Voronoi Diagram (VD), is presented in Algorithm 1.

TL_{max} and TL denote the maximum tessellation level and the current tessellation level respectively, T_1 is the threshold value for edge length; \mathcal{N} represents the set of constraint nucleation points for f , which are projections of door nodes, concave corner nodes, and staircase nodes of f ; \mathcal{R} denotes the set of VD cells; \mathcal{M} is the set of cells for testing, in which each cell is added from \mathcal{R} and includes node N_j ; edge (j, k) is bounded by the intersection points in \mathcal{I} , which divides R_k into two cells such that each cell includes one of the nodes N_j and N_k ; e denotes the edge connecting nucleation points of adjacent cells in \mathcal{R} ; constraint edges are the original polygon edges of f .

Fig. 4 illustrates a close-up of a single level, the second floor, on a real-word passenger ship, “Yangtze Gold 7”. Feasible routes for passengers, generated after the first iteration of tessellation and the second iteration of tessellation, are represented in Fig. 4b and Fig. 4c respectively.

C. Navigation model parameter estimation.

1) *Estimation of Worst-case Delay and Typical Delay:* Table II shows the walking speeds of an evacuee when the distance from the front evacuee is more than 0.5 m, which are regulated by MSC/Circ.1238 [38]. According to the speed data, we assume that the initial typical speed across each segment is 1.0 m/s, 0.8 m/s, or 1.2m/s, depending on which type of facility the segment is located in. One segment includes only one type of facility. Considering the effect of the continuously increasing heel angle of a damaged ship on pedestrian movement, passengers’ typical speed will dynamically updated. Based on [39], it is found that passengers’ age and gender have a trivial impact on their speed at sea. Therefore, in this paper we only utilize the current slope angle of a passenger ship and the average individual walking speed to determine the dynamic

Algorithm 1: Navigation Network Construction Algorithm

Input: Floor polygon f , TL_{max} , T_1 ;

Output: Tessellation f ;

```

1  $\mathcal{R}_1 = f$ ;
2  $TL = 1$ ;
3 while  $TL \leq TL_{max}$  do
4   for each element  $N_j$  in  $\mathcal{N}$  do
5     for each element  $R_k$  in  $\mathcal{M}$  do
6       Calculate a bisector line  $bl$  of  $\overline{N_j N_k}$ ;
7       Calculate the set  $\mathcal{I}$  of the intersection
         points of  $bl$  with  $R_k$ ;
8       if  $|\mathcal{I}| \geq 2$  then
9         Create edge  $(j, k)$  and insert it into the
           existing tessellation of the polygon  $f$ ;
10        Add the cell including  $N_j$  to  $\mathcal{R}$ ;
11        Add the cells adjacent to the edges
          intersected by  $(j, k)$  to  $\mathcal{M}$ ;
12      end
13    end
14    Remove edges from inside the boundary of the
      newly created cell for  $N_j$ .
15  end
16  for each edge  $e$  in the tessellation of  $f$  do
17    if  $|e| \geq T_1$ , or  $e$  is bounded by two constraint
      nucleation points, or end-points of  $e$  lie on
      non-collinear constraint edges then
18      Calculate the bisector point  $bp$  for  $e$  and
        add  $bp$  into  $\mathcal{N}$ ;
19    end
20  end
21   $TL = TL + 1$ ;
22 end

```

typical speed. Combining this speed and segment length gives rise to the typical delay of each segment.

Considering the influence of watertight bulkheads on liquid flow in ship compartments, two different procedures are used to estimate worst-case speed across different segments: For segments which are not affected by flooding water due to watertight bulkheads, the worst-case speed across them can be calculated as follows:

$$S_W = \begin{cases} 0.55 \times R_1^{30^\circ} & \text{Stairs (down)} \\ 0.44 \times R_1^{30^\circ} & \text{Stairs (up)} \\ 0.67 \times R_1^{30^\circ} & \text{Corridors} \\ 1.2 \times R_1^{30^\circ} & \text{Open space} \end{cases} \quad (2)$$

where $R_1^{30^\circ}$ denotes the reduction factor of walking speed at heel angle 30° ; for segments affected by flooding water, the worst-case speed is as followed:

$$S_W = \begin{cases} 0.55 \times R_1^{30^\circ} \times R_2^{0.8} & \text{Stairs (down)} \\ 0.44 \times R_1^{30^\circ} \times R_2^{0.8} & \text{Stairs (up)} \\ 0.67 \times R_1^{30^\circ} \times R_2^{0.8} & \text{Corridors} \\ 1.2 \times R_1^{30^\circ} \times R_2^{0.8} & \text{Open space} \end{cases} \quad (3)$$

where we take the effect of flooding water on pedestrian

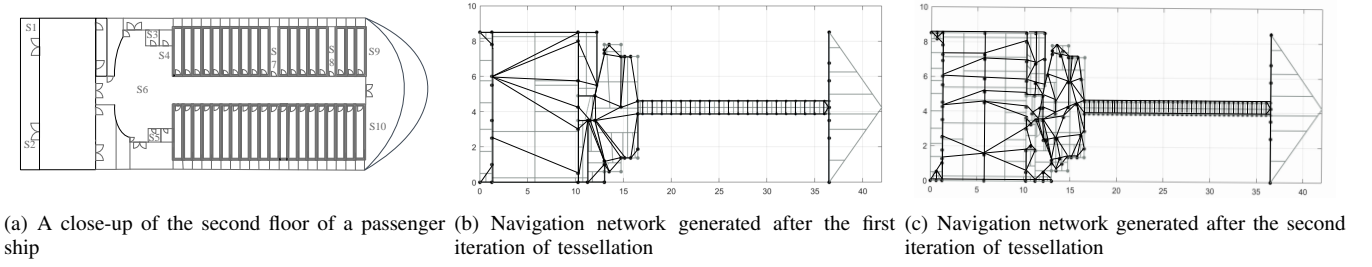


Fig. 4: Navigation network of the second floor of a passenger ship

movement into account, $R_2^{0.8}$ represents the speed reduction factor at water depth 0.8 m [40].

TABLE II: VALUES OF SPECIFIC FLOW AND SPEED

Facility Type	Flow (p/(ms))	speed of persons (m/s)
Stairs (down)	0	1.0
	0.54	1.0
	1.1	0.55
Stairs (up)	0	0.8
	0.43	0.8
	0.88	0.44
Corridors	0	1.2
	0.65	1.2
	1.3	0.67

2) *Estimation of $\mathcal{D}_h^v(t)$* : The default landmark role is general landmark. In the initialization phase, each element in $\mathcal{D}_h^v(t_0)$ is set as follows:

$$D_h^{\vec{v}_i \vec{v}_j}(t_0) = \begin{cases} +\infty, & \text{if } v_j \neq v_e \\ T_c(t_0), & \text{if } v_j = v_e \end{cases} \quad (4)$$

Where $T_c(t_0)$ denotes the remaining ship survival time at time t_0 . The interval of time from hazardous landmarks to segments may be captured in a real-time pattern using the periodic probes. A landmark v_j receiving a probe from a hazardous landmark determines whether to update the value in $\mathcal{D}_h^v(t)$ based on Algorithm 1 in [34].

D. Path generation algorithm.

This section describes our proposed ϵ -approximation algorithm for emergency navigation in ship indoor environments. We first briefly describe two basic techniques termed *rounding and scaling* and *relaxing* used in the algorithm.

1) *Rounding and scaling*: To approximate p^* in polynomial time, we perform the *rounding and scaling* operation as shown in Algorithm 2. The complexity of Algorithm 2 is $O(|\mathcal{E}| \frac{U}{L} \frac{n}{\epsilon})$, where U and L denote the upper and lower bound for the optimal solution respectively, n is the number of landmarks, and ϵ ($0 < \epsilon < 1$) is an approximation factor.

$g_j(d_T)$ denotes the minimum worst-case delay of a u - j path whose typical time is at most d_T .

Algorithm 2: Rounding and Scaling Algorithm

Input: $\mathcal{G}=(\mathcal{V}, \mathcal{E})$, $\{d_W^{\vec{v}_i \vec{v}_j}, d_T^{\vec{v}_i \vec{v}_j}\}_{\vec{v}_i \vec{v}_j \in \mathcal{E}}$, $T_c(t)$, L , U , ϵ ;

Output: \hat{p}^* and \hat{d}_T^p ;

```

1 for each  $\vec{v}_i \vec{v}_j \in \mathcal{E}$  do
2    $\hat{d}_T^{\vec{v}_i \vec{v}_j} = \lfloor \frac{d_T^{\vec{v}_i \vec{v}_j} (n-1)}{L\epsilon} \rfloor + 1$ 
3 end
4  $\hat{U} = \lfloor \frac{U(n-1)}{L\epsilon} \rfloor + n$ ;
5 for  $j \neq u$  do
6    $g_j(0) = +\infty$ 
7 end
8 for  $\hat{d}_T = 0, 1, 2, \dots, \hat{U}$  do
9    $g_u(\hat{d}_T) = 0$ ;
10 end
11 for  $\hat{d}_T = 1, 2, 3, \dots, \hat{U}$  do
12   while  $j \neq u$  do
13     relaxing  $g_j(\hat{d}_T)$ 
14   end
15   if  $g_e(\hat{d}_T) \leq T_c(t)$  then
16      $\hat{d}_T^p = \hat{d}_T$ ;
17     return  $\hat{d}_T^p$  and  $\hat{p}^*$ ;
18     break;
19   end
20 end
```

2) *Relaxing*: The *relaxing* operation is as follows:

$$g_j(d_T) = \min\{g_j(d_T - 1), \min_{k | d_T^{\vec{v}_k \vec{v}_j} \leq d_T} \{g_k(d_T - d_T^{\vec{v}_k \vec{v}_j}) + d_W^{\vec{v}_k \vec{v}_j}\}\}, \quad (5)$$

In the following, we present the overall structure of our approximation algorithm.

3) *Top-level algorithm*: In reality, hazardous regions and typical delay may vary in time as mentioned in Section II-B and Section III-A. The effects of such dynamics on Algorithm 2 mainly reflect in two aspects. First, the path provided by Algorithm 2 may not be optimal any more due to the change of typical delay. Second, the path may be blocked because the encroachment of hazards upon segments, which inevitably results in user oscillations and thus keeps the user remaining in danger for a longer period of time and finally misses the hard

deadline for ship evacuation. In order to compensate for these impacts, we design Algorithm 3 which informs each passenger about a hazard-avoid oscillation-free route to reach the exit landmark within a given deadline under all circumstances, while guaranteeing to get the passenger to experience a no more than $(1+\epsilon)d_T^{p^*}(\mathcal{T})$ typical delay. In the following, we present the details of each step of Algorithm 3:

- Line 2, in the variable $\widehat{p^*}(t)$, computes a $T_c(t)$ -path from v_u to v_e with a no more than $(1+\epsilon)d_T^{p^*}(\mathcal{T})$ typical delay.
- In Line 3, a variable *test1* is set equal to FALSE, thereby indicating that there are not segments in the path $\widehat{p^*}(t)$, which a user cannot cross before hazards encroach on them.
- Line 4 constructs a set \mathcal{E}_{test} to subsume segments that a user cannot cross before hazards damage them along the planned path.
- The for loop of Lines 5-10 check each segment already in path $\widehat{p^*}(t)$ to determine whether it should be subsumed into \mathcal{E}_{test} . In Line 7, $\widehat{p_{ij}^*}(t)$ represents a connected sequence of segments from v_u to v_j , which are components of $\widehat{p^*}(t)$.
- Lines 11-14 remove the segments in set \mathcal{E}_{test} from \mathcal{E} in order to reconstruct the navigation network connectivity and call Algorithm 2 based on the current $\mathcal{G}=(\mathcal{V}, \mathcal{E})$ to get path $\widehat{p^*}(t)$ and select the neighbor landmark $\hat{\pi}_u$ on path $\widehat{p^*}(t)$ as the user's next landmark. During the next iteration, $\hat{\pi}_u$ is set as the new user landmark as shown in Line 15.
- Line 16 checks whether the typical delay has changed. According to the result, Lines 17-21 perform in the following manner:
 - 1) If there is a change in typical delay then we recalculate the path $\widehat{p^*}(t_{new})$ from the current user landmark to exit landmark according to the changed typical delay, where t_{new} is the timestamp of arriving at the current landmark;
 - 2) If the typical delay has not changed then the user moves to the neighbor landmark of the current user landmark on the path $\widehat{p^*}(t_{N-1})$, where t_{N-1} is the timestamp of departing from the previous landmark.

E. An example of WEND

Let us now consider the path generation of the example graph of Fig. 5a by the algorithm of Section IV-D above. The blue circle represents the user landmark, the black circles denote general landmarks, the red circle represent the hazardous landmark, and the green circle indicates the exit landmark. Each segment is characterized by two delay parameters: the typical delay $d_T^{v_i v_j}(t)$ (see the blue numbers), and the worst-case delay $d_W^{v_i v_j}$ (see the red numbers). The red arrow indicates the moving direction of the emergency site. Suppose that v_1 becomes a hazardous landmark at time $t = 30$. In addition, it is assumed that $T_c(t_0)$ and ϵ respectively equal 80 and 0.25.

1) Compute upper and lower bounds of the optimal path.

According to Algorithm 4, the upper bound of the optimal path $p^*(\mathcal{T})$ is set to $5+5+14+5+14+12.5+5=60.5$. As shown in subgraph (b), $\mathcal{G}^7 = (\mathcal{V}, \mathcal{E}^7(t_0))$ has a 80-path p^7 and all

Algorithm 3: Top-level Algorithm

Input: $\mathcal{G}=(\mathcal{V}, \mathcal{E})$;
Output: $p^*(\mathcal{T})$;

```

1 while  $v_u \neq v_e$  do
2   Call procedure Rounding and Scaling Algorithm;
3    $test1 = \text{FALSE}$ ;
4    $\mathcal{E}_{test} = \{\}$ ;
5   for each segment  $\overrightarrow{v_i v_j} \in \widehat{p^*}(t)$  do
6     if  $d_W^{p_{ij}^*}(t) > D_h^{\overrightarrow{v_i v_j}}(t)$  then
7        $test1 = \text{TRUE}$ ;
8       insert  $\overrightarrow{v_i v_j}$  into  $\mathcal{E}_{test}$ 
9     end
10  end
11  if  $\mathcal{E}_{test} \neq \{\}$  then
12    Set  $\mathcal{E} = \mathcal{E} \setminus \mathcal{E}_{test}$ ;
13    Go to Line 2;
14  end
15   $v_u = \pi_u$ ;
16  if there is a change in typical delay then
17    Go to Line 2;
18  else
19     $v_u = \pi_u$ ;
20    Go to Line 16;
21  end
22 end

```

segments in p^7 belong to $\mathcal{E}_s^{nh}(p^7)$, and in \mathcal{G}^6 (see subgraph (c)) all paths from v_u to v_e have worst-case delays larger than 80. Then the lower bound equals $d_T^7(t_0)=10$.

2) **Call procedure Rounding and Scaling Algorithm.** Upon calling the procedure Rounding and Scaling Algorithm, $\widehat{p^*}(t_0) = \langle v_u, v_0 \rangle, v_8, v_7, v_6, v_5, v_9, v_e \rangle$, an 80-path from v_u to v_e with a no more than $(1+0.25)d_T^{p^*}(t_0)$ typical delay, is obtained as shown in subgraph (d).

3) **Check segments of path $\widehat{p^*}(t_0)$.** According to the moving direction and speed of the emergency site, we check all segments of path $\widehat{p^*}(t_0)$ and obtain the set $\mathcal{E}_{test}=\{\}$. That is, all segments in $\widehat{p^*}(t_0)$ belong to safe segments for $\widehat{p^*}(t_0)$.

4) **User navigation guidance and path replanning.** We set v_8 as the next landmark. Suppose that it takes 20 to arrive at v_8 . That is, the duration remaining between the current instant and the instant by which the user must reach the exit is 60. Upon reaching v_8 , we assume that it checks there are changes in typical delays as shown in subgraph (e). Invoke the procedure Rounding and Scaling again. An updated optimal path $\widehat{p^*}(t=20) = \langle v_u, v_8 \rangle, v_7, v_{10}, v_9, v_e \rangle$ is calculated, and thus v_7 is set as the next landmark. The process of user navigation guidance and path replanning is repeated until the user is guided to the exit. In this example, there are no variations in typical delays when the user gets to v_7, v_{10} , and v_9 . Therefore, $p^*(\mathcal{T}) = \langle v_0, v_8, v_7, v_{10}, v_9, v_e \rangle$.

V. ANALYSIS AND DISCUSSIONS

In this section, we compute an upper bound and a lower bound on the optimal route defined in Section III-B, and

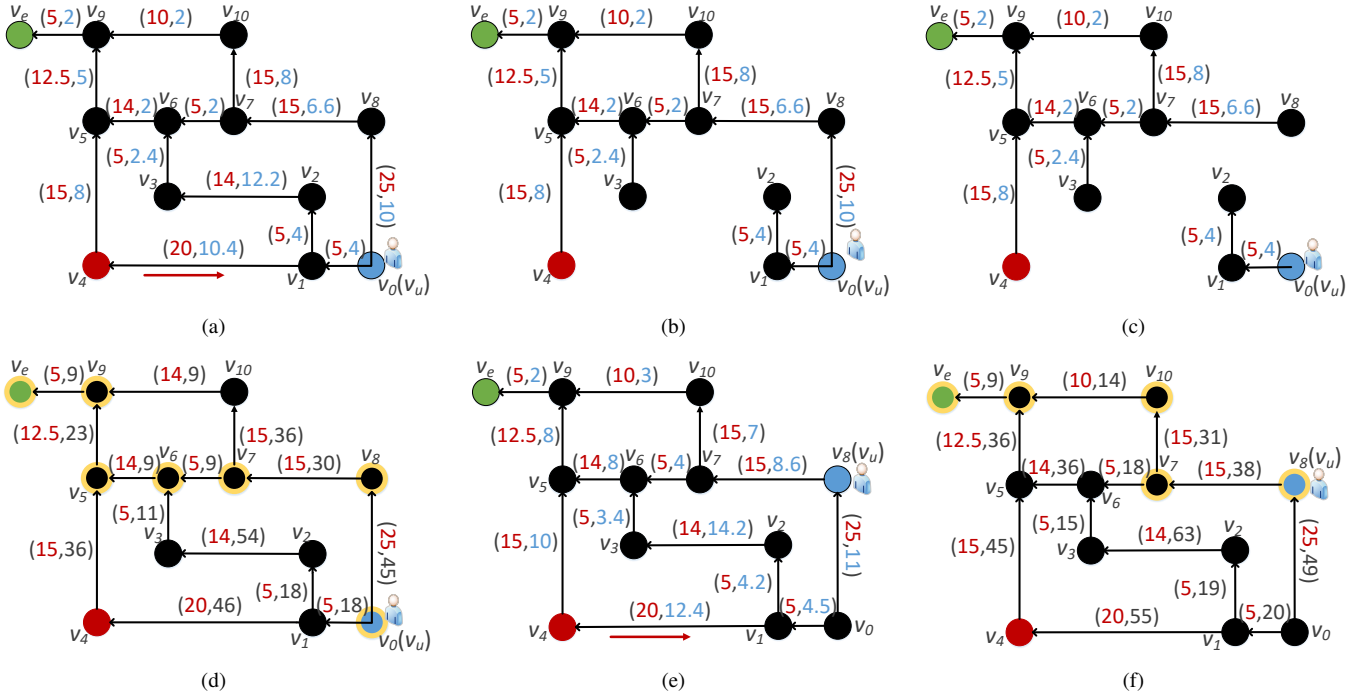


Fig. 5: Example graph. (a) The navigation model at the initial time t_0 , $\mathcal{G}=(\mathcal{V}, \mathcal{E}(t_0))$. A user is located at the landmark v_0 (b) The graph $\mathcal{G}^7=(\mathcal{V}, \mathcal{E}^7(t_0))$. (c) The graph $\mathcal{G}^6=(\mathcal{V}, \mathcal{E}^6(t_0))$. (d) The graph after rounding and scaling each segment $\overrightarrow{v_i v_j}$ in $\mathcal{G}=(\mathcal{V}, \mathcal{E}(t_0))$, $p^*(t_0) = \langle v_u, v_8, v_7, v_6, v_5, v_9, v_e \rangle$. (e) The navigation model at time $t=20$, $\mathcal{G}=(\mathcal{V}, \mathcal{E}(t=20))$. The user moves to the landmark v_8 (f) The graph after rounding and scaling each segment $\overrightarrow{v_i v_j}$ in $\mathcal{G}=(\mathcal{V}, \mathcal{E}(t=20))$, $p^*(t=20) = \langle v_u, v_7, v_{10}, v_9, v_e \rangle$.

then calculate the error bound of our proposed approximate solution. Finally, the analysis of the impact of the correlation of typical delay and worst-case delay is presented.

A. Upper/lower bounds on our defined optimal path.

Here we present how to calculate the high-quality upper and lower bounds of the optimal path $p^*(\mathcal{T})$.

1) *Upper bound of $p^*(\mathcal{T})$:* First, we calculate a route p^U with the minimum worst-case delay based on the navigation graph model without hazardous regions inserted. The minimum worst-case delay is an upper bound of the path $p^*(\mathcal{T})$. Then, we gradually insert hazardous regions one by one to update the upper bound, so that it can bound $p^*(\mathcal{T})$ more accurately. Algorithm 4 shows the pseudo code of the estimation of the upper bound of $p^*(\mathcal{T})$. \mathcal{R}_h denotes the set of hazardous regions, n_h is the total number of hazardous regions, $\mathcal{G}_s^i = (\mathcal{V}, \mathcal{E}_s^i(p^U))$, $\mathcal{E}_s^i(p^U)$ denotes the safe segments for p^U after the i 'th hazardous region is inserted. A segment $\overrightarrow{v_i v_j}$ can be regarded as a safe segment for p^U if and only if $d_W^{p_{uj}} \leq D_h^{\overrightarrow{v_i v_j}}(t_0)$, $p_{uj} \in p^U$.

2) *Lower bound of $p^*(\mathcal{T})$:* First, we start by ordering the distinct values $d_T^{\overrightarrow{v_i v_j}}(t_0)$. Let $d_T^1(t_0) < d_T^2(t_0) < \dots < d_T^l(t_0)$ be all the distinct typical delay values of the segments. Clearly $l \leq |\mathcal{E}| = m$. Define $\mathcal{E}^i(t_0)$ to be the set of segments with typical delay not greater than $d_T^i(t_0)$. That is, for $1 \leq i \leq l$, $\mathcal{E}^i(t_0)$ is defined as follows:

$$\mathcal{E}^i(t_0) \triangleq \left\{ \overrightarrow{v_i v_j} \in \mathcal{E} \mid d_T^{\overrightarrow{v_i v_j}}(t_0) \leq d_T^i(t_0) \right\}$$

Algorithm 4: Calculating Upper Bound of $p^*(\mathcal{T})$

Input: $\mathcal{G}=(\mathcal{V}, \mathcal{E})$, \mathcal{R}_h ;

Output: U ;

- 1 Calculate U (corresponding to the minimum worst-case delay from initial user landmark to exit landmark) based on \mathcal{G}_s^0 ;
- 2 **for** $i \leftarrow 1$ to n_h **do**
- 3 **for each segment** $\overrightarrow{v_i v_j} \in p^U$ **do**
- 4 **if** $\overrightarrow{v_i v_j} \notin \mathcal{E}_s^i(p^U)$ **then**
- 5 Update the path p^U based on \mathcal{G}_s^i ;
- 6 **break**;
- 7 **end**
- 8 **end**
- 9 **end**

Let $\mathcal{G}^i = (\mathcal{V}, \mathcal{E}^i(t_0))$, then $\mathcal{G}^l = \mathcal{G}$. Moreover, since \mathcal{G}^l must have a $T_c(t_0)$ -path p^l and all segments in p^l belong to $\mathcal{E}_s^{n_h}(p^l)$, (otherwise the instance has no solution) there must exist a unique index j ($1 \leq j \leq l$) such that \mathcal{G}^j has a $T_c(t_0)$ -path p^j and all segments in p^j belong to $\mathcal{E}_s^{n_h}(p)$, and in \mathcal{G}^{j-1} all paths \mathcal{P}^{j-1} from initial landmark to exit landmark (for $\forall p^{j-1} \in \mathcal{P}^{j-1}, \forall \overrightarrow{v_i v_j} \in \mathcal{E}_s^{n_h}(p^{j-1})$) have worst-case delay larger than $T_c(t_0)$. Then

$$L = d_T^j(t_0) \leq d_T^{p^*}(\mathcal{T})$$

B. Error bound of our proposed approximate solution.

Here we have discussion on the error bound caused by our approximate algorithm. First, we estimate the error bound when the typical delay is static, then based on the results in Lemma 1, we give the error bound when typical delay varies with time.

Lemma 1. For the situation where the typical delay is static: If $U \geq d_T^p$, then Algorithm 2 returns a feasible path, p^* , that satisfies $d_T^{p^*} \leq d_T^p \leq d_T^{p^*} + L\epsilon$.

Proof. By definition, $d_T^{p^*} \leq d_T^p$. For each segment $\overrightarrow{v_i v_j} \in p^*$, $\widehat{d_T^{p^*}} \leq \frac{d_T^{p^*}(n-1)}{L\epsilon} + 1$. Thus,

$$\begin{aligned} \widehat{d_T^{p^*}} &\triangleq \sum_{\overrightarrow{v_i v_j} \in p^*} \widehat{d_T^{p^*}} \leq \frac{(n-1)}{L\epsilon} \sum_{\overrightarrow{v_i v_j} \in p^*} d_T^{p^*} + n - 1 \\ &\leq \frac{(n-1)U}{L\epsilon} + n - 1 \leq \hat{U} \end{aligned} \quad (6)$$

Each segment $\overrightarrow{v_i v_j} \in p$ satisfies:

$$\frac{d_T^{p_i v_j}(n-1)}{L\epsilon} \leq \widehat{d_T^{p_i v_j}} \leq \frac{d_T^{p_i v_j}(n-1)}{L\epsilon} + 1. \quad (7)$$

Thus,

$$\begin{aligned} d_T^p &\triangleq \sum_{\overrightarrow{v_i v_j} \in p} d_T^{p_i v_j} \leq \frac{L\epsilon}{n-1} \sum_{\overrightarrow{v_i v_j} \in p} \widehat{d_T^{p_i v_j}} \triangleq \frac{L\epsilon}{n-1} \widehat{d_T^p} \\ &\leq \sum_{\overrightarrow{v_i v_j} \in p} d_T^{p_i v_j} + L\epsilon \triangleq d_T^p + L\epsilon \end{aligned} \quad (8)$$

According to Equation (6) and Equation (8):

$$d_T^{p^*} \leq \frac{L\epsilon}{n-1} \widehat{d_T^{p^*}} \leq \frac{L\epsilon}{n-1} \widehat{d_T^p} \leq d_T^p + L\epsilon \quad (9)$$

Lemma 2. For the situation where the typical delay is dynamic: If $U \geq d_T^p(\mathcal{T})$ then Algorithm 3 returns a feasible path, $p^*(\mathcal{T})$, that satisfies $d_T^{p^*}(\mathcal{T}) \leq d_T^p(\mathcal{T}) \leq d_T^{p^*}(\mathcal{T}) + L\epsilon$.

Proof. By definition, $d_T^{p^*}(\mathcal{T}) \leq d_T^p(\mathcal{T})$. We assume that a user traverses m_1 segments after the first change on typical delay. According to Equation (7):

$$\begin{aligned} \sum_{\overrightarrow{v_i v_j} \in p_{un_1}} d_T^{p_i v_j} &\leq \frac{L\epsilon}{n-1} \sum_{\overrightarrow{v_i v_j} \in p_{un_1}} \widehat{d_T^{p_i v_j}} \\ &\leq \sum_{\overrightarrow{v_i v_j} \in p_{un_1}} d_T^{p_i v_j} + \frac{m_1 L\epsilon}{n-1} \end{aligned} \quad (10)$$

where p_{un_1} denotes the route from v_u to v_{n_1} , which consists of m_1 segments. n_i denotes the landmark at which a user arrives after traversing m_1 segments, and m_i is the number of segments that a user travels after the i 'th change minus the number of the already-traversed segments after the $(i-1)$ 'th change. We assume that the typical delay changes N_c times until arriving at the exit landmark v_e . The overall delay of a

path thus:

$$\begin{aligned} d_T^p(\mathcal{T}) &\triangleq \sum_{i=0, j=1}^{i=N_c-1, j=N_c} \sum_{\overrightarrow{v_i v_j} \in p_{n_i n_j}} d_T^{p_i v_j}(\Delta t) \\ &\leq \frac{L\epsilon}{n-1} \sum_{i=0, j=1}^{i=N_c-1, j=N_c} \sum_{\overrightarrow{v_i v_j} \in p_{n_i n_j}} \widehat{d_T^{p_i v_j}}(\Delta t) \\ &\triangleq \frac{L\epsilon}{n-1} \widehat{d_T^p}(\mathcal{T}) \leq \sum_{i=0, j=1}^{i=N_c-1, j=N_c} \sum_{\overrightarrow{v_i v_j} \in p_{n_i n_j}} d_T^{p_i v_j}(\Delta t) \\ &\quad + \frac{(m_1 + \dots + m_{N_c})L\epsilon}{n-1} \leq d_T^p(\mathcal{T}) + L\epsilon \end{aligned} \quad (11)$$

where n_0 and n_{N_c} denote the user landmark v_u and the exit landmark v_e , respectively. According to Equation (6):

$$\begin{aligned} \widehat{d_T^{p^*}} &\triangleq \sum_{i=0, j=1}^{i=N_c-1, j=N_c} \sum_{\overrightarrow{v_i v_j} \in p_{n_i n_j}^*} \widehat{d_T^{p_i v_j}} \\ &\leq \frac{(n-1)}{L\epsilon} \sum_{i=0, j=1}^{i=N_c-1, j=N_c} \sum_{\overrightarrow{v_i v_j} \in p_{n_i n_j}^*} d_T^{p_i v_j} \\ &\quad + m_1 + m_2 + \dots + m_{N_c} \\ &\leq \frac{(n-1)U}{L\epsilon} + n - 1 \leq \hat{U} \end{aligned} \quad (12)$$

According to Equation (11) and Equation (12):

$$d_T^{p^*}(\mathcal{T}) \leq \frac{L\epsilon}{n-1} \widehat{d_T^{p^*}}(\mathcal{T}) \leq \frac{L\epsilon}{n-1} \widehat{d_T^p}(\mathcal{T}) \leq d_T^p(\mathcal{T}) + L\epsilon \quad (13)$$

C. Impact of the correlation of typical delay and worst-case delay.

In this part, we analyze the impact of the correlation of typical delay and worst-case delay on our path planning algorithm. In our scenario, the correlation between typical delay and worst-case delay is neither perfect positive nor perfect negative, and thus the algorithm needs to build off the ideas of the restricted shortest path algorithm. However, if the correlation coefficient between the two delay parameters is +1.0 or -1.0, various shortest path algorithm [41] may be applied to calculate the hazard-avoid $T_c(t)$ -path with the minimum typical delay only utilizing the worst-case delay on each segment. Actually, in such a case the path with the minimum worst-case delay from the initial user landmark to exit landmark is the defined optimal solution.

VI. PERFORMANCE EVALUATION

We carry out extensive simulations to evaluate the performance of our approach. We compare this algorithm with the state-of-the-art navigation approaches, namely OPEN and ANT, from five perspectives, i.e., average escape time, navigation success ratio, average path length, minimum distance to hazards, and emergency response time.

A. Experimental Scenario & System Flowchart

Fig. 6 shows the flow diagram of the proposed navigation scheme. The current location of the passenger is detected using the received strength of wireless signals on his/her smartphone. The sensor with the strongest signal strength serves as the user node, and the user node information is transmitted to the path planning server. In addition, the location information on each sensor is also sent to the server and used to construct the 3D topological model and predict the typical delay and worst-case delay on segments. In case of an emergency, the WSN forwards the information of hazards to the server, and the path generator is triggered to compute the route. The trapped user can be guided to the next landmark by complying with the indicator on the device. In addition, the guidance direction can be updated in a real-time pattern based on the dynamics of ship inclination.

To assess the performance of WEND on a damaged passenger ship, we conduct our proposed scheme in navigation scenarios that model the topological structure of a real-world cruise ship, namely the "Yangtze Gold 7" cruise. To examine the scalability, we vary the number of iterations of tessellation to generate navigation networks with different numbers of landmarks. We define two different navigation scenarios as shown in Fig. 4: (i) navigation network generated after the first iteration of tessellation (named SF) where 98 nodes are deployed (see Fig. 4b), (ii) navigation network generated after the second iteration of tessellation (named SS) where 196 nodes are deployed (see Fig. 4c). "Yangtze Gold 7" passenger ship can carry 398 passengers. Therefore, the number of evacuated users ranges from 30 to 390 in our simulations. For each trial of the user count, we insert three hazardous regions into the network and allocate hazards' initial size and expansion (i.e., expanding direction and speed). Hazard is assumed to exhibit dynamics in only one pattern, i.e., expansion, in our simulation. During the simulations, the ratio of the size of each hazardous region to the total network size is maintained below 5%, the same with that assumed in [14]. The locations of emergency sites and users are randomly generated in the field for each round, and all users are required to arrive at the same exit. For each trial, we exploit the navigation scenario (i) unless explicitly stated otherwise.

Navigation model parameters including worst-case delay and initial typical delay on each segment are set using parameters of a real-world passenger ship, "Yangtze Gold 7" passenger ship. The hard deadline for ship evacuation and dynamics of typical delay are determined by the analysis of maritime accident investigation reports and case studies. In our simulations, the worst-case and initial typical moving speed of a user are set as 0.1088 meters and 0.4143 meters per second, respectively. We assume that the typical speed changes every 10 s and the deadline for evacuation is fixed at 100 s. The simulation results reported below are the average values after 20 runs.

We perform the simulation experiments written in Matlab on a personal computer (Operating System: Windows 10 Education) equipped with Intel Core i5-8400 CPU @ 2.8GHz and 8.0 GB memory.

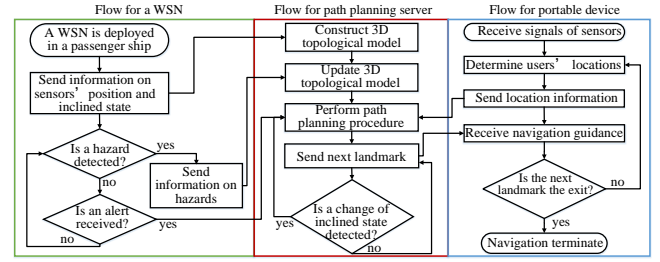


Fig. 6: Flow diagram of our navigation scheme

B. Average escape time

This group of simulations evaluates path efficiency by comparing the user escape time of the route planned in each approach. A shorter escape time indicates a more efficient navigation path, as users are more likely to evacuate to the exit within the deadline along this route.

Fig. 7 shows the user escape time of the three approaches under different values of ϵ for navigation scenario (i) that has one user node deployed randomly. The data is collected from 35 trials. We can see that our method outperforms OPEN and ANT by yielding a shorter escape time in more than 90% and 70% of trials, respectively. The escape time provided by OPEN is much longer than that offered by our approach and ANT in some cases, such as Experiment 3 in Fig. 7a and Experiment 14 in Fig. 7b. That is because the OPEN scheme aims at generating a navigation path that minimizes the probability of oscillation. It proposes a novel path metric, ENO, calculated only by the spatial and temporal accumulated emergency level to achieve an oscillation-free path. Without sufficient consideration of the typical delay on each segment, OPEN is likely to result in large detours and thus long escape time, and finally missing the hard deadline for evacuation. We also notice that in rare cases, such as Experiment 12 in Fig. 7b, it takes longer to escape along the navigation route of our approach and ANT. That is because two delay parameters are specified on each segment in the WEND scheme, and thus the provided path not only has the shortest typical time but also guarantees the user's arrival within the deadline considering worst-case delays. That is, it is not the theoretically fastest safe path. In addition, ANT shows equal or better performance compared with WEND when the escaping time of the user is less than 10 s. It is mainly because we assume that the typical speed changes every 10 s. That is to say, there is no dynamic variation when the escaping time is no greater than 10 s. In this case, ANT can provide the optimal guidance $p^*(\mathcal{T})$, and the path provided by WEND, an approximation scheme for the ship emergency navigation, can achieve a no more than $(1+\epsilon)d_T^p(\mathcal{T})$ typical delay.

Fig. 8 shows the average escape time of each approach under different numbers of users. In the figure, experimental results for $\epsilon = 1$, $\epsilon = 0.5$, $\epsilon = 0.25$ are shown. We can see that in average WEND decreases the navigation time by 20%, 35%, 50% compared with OPEN, and 4%, 12%, and 20% compared with ANT, for $\epsilon = 1$, $\epsilon = 0.5$, and $\epsilon = 0.25$ respectively. This result demonstrates the superior navigation efficiency using WEND when ϵ is set as 0.5 and 1. WEND can determine a

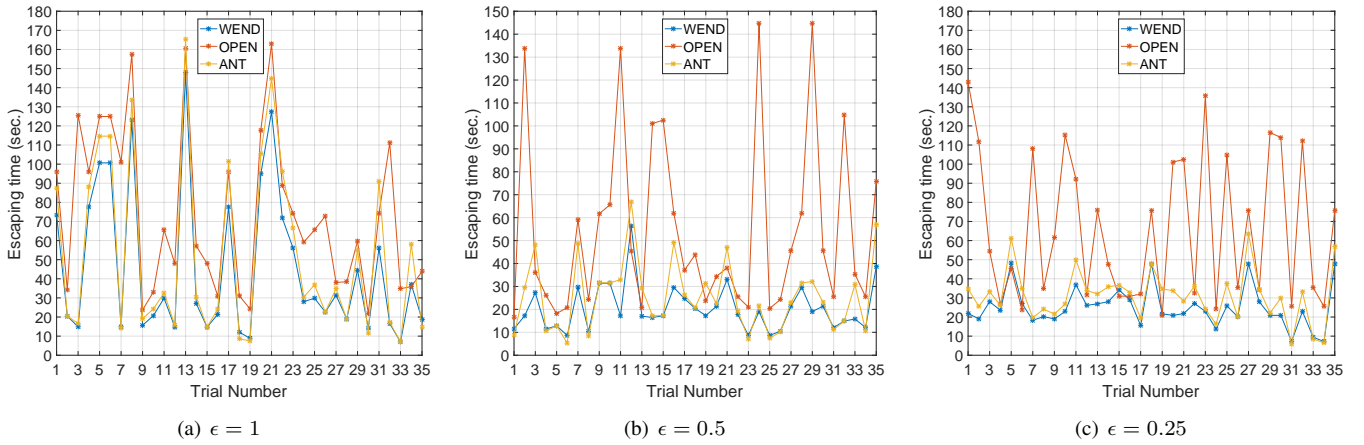


Fig. 7: Comparisons of average escape time for navigation scenario (i) under different values of ϵ

navigation path with a no more than $(1+\epsilon)d_T^p(\mathcal{T})$ typical delay while simultaneously considering the hazard motion tendency and the worst-case source-to-destination delay guarantee.

We further evaluate the scalability of WEND at larger scales. Specifically, we carry out a group of simulations, where 40, 90, ..., 340, 390 users are inserted in navigation scenarios (i) and (ii), respectively. Fig. 9 shows the average escape time of OPEN, ANT, and WEND. OPEN98, ANT98, and WEND98 indicate the results in scenario (i) where 98 nodes are deployed, and OPEN196, ANT196, and WEND196 denote the experiments conducted in scenario (ii) with 196 nodes deployed. We can observe that all three methods have increased ratios as the network size is increased. That is probably because distance-based delay measurement in the two approaches becomes more accurate when more nodes are involved as the network scale increases. However, WEND outperforms both ANT and OPEN by always taking shorter to navigate users in both scenario (i) and scenario (ii).

C. Navigation success ratio

We evaluate the navigation success ratio of the three methods in this group of simulations, which is measured by the average arrival possibility within the deadline for evacuation. The time available for passenger evacuation on a damaged ship is limited. Passengers must arrive at a muster station within the deadline; otherwise, their navigation fails. Therefore, we can evaluate the navigation success ratio by calculating the possibility of reaching the exit within the deadline. In our simulation, this possibility is expressed as $1 - \frac{F(d_T^p(\mathcal{T}))}{T_c(t_0)}$, where $F(d_T^p(\mathcal{T}))$ is computed by (14).

$$F(d_T^p(\mathcal{T})) = \begin{cases} 0, & \text{if } d_T^p(\mathcal{T}) \leq T_c(t_0) \\ d_T^p(\mathcal{T}) - T_c(t_0), & \text{if } d_T^p(\mathcal{T}) > T_c(t_0) \end{cases}, \quad (14)$$

In the controlled simulation, we consider that missing the deadline is the only factor, which causes navigation failure.

We inject 150 user nodes into the navigation scenario (i). ϵ is set to 1, 0.5, and 0.25. Fig. 10 shows the CDF of the arrival possibility within the deadline of OPEN, ANT, and WEND. We can see that more than 70, 80, and 85 percent

of users can be successfully navigated to the exit within the deadline for evacuation using WEND for $\epsilon = 1$, $\epsilon = 0.5$, and $\epsilon = 0.25$ respectively. All of them are much better than the cases with OPEN which can guarantee that around 60% of users can be successfully navigated. OPEN fails to ensure navigation success in certain scenarios because it is likely to cause detours, which increases user escape time and thus leads to navigation failure. Moreover, WEND's performance clearly outperforms ANT because ANT does not update the look-up tables according to the changing typical delay.

D. Minimum distance to hazards

We evaluate the effectiveness of WEND in terms of the absolute safety of the planned path, which is measured by the minimum distance from the route to hazardous regions. A larger distance to hazards indicates a better safety of the navigation route, as it provides a better chance for the guided user to safely bypass hazardous regions. Fig. 11a shows the minimum distance to hazards of the three approaches under different numbers of users. We can see that the OPEN approach presents the superior result with the distance more than five meters. WEND and ANT have the distance of approximately four meters, which is slightly lower than that of OPEN, however, which does not indicate a considerable threat to users. It is unnecessary to find a path for which the optimal result with the maximum safe distance is achieved because that means the navigation decision is often over-conservative, which can increase the time spent in the navigation environment, thereby reducing the overall safety of guided users. In comparison, WEND greatly reduces the stay time of users in the hazardous environment, enhancing the opportunity to arrive at the exit within the specified deadline.

E. The impact of deadline

We emphasize that the ability to deal with different deadlines is one of the core advantages of our method. To test the impact of different settings of the deadline, we conduct simulations on the navigation scenario (i) for $\epsilon = 0.5$ with different deadlines. As shown in Fig. 11b, for WEND and

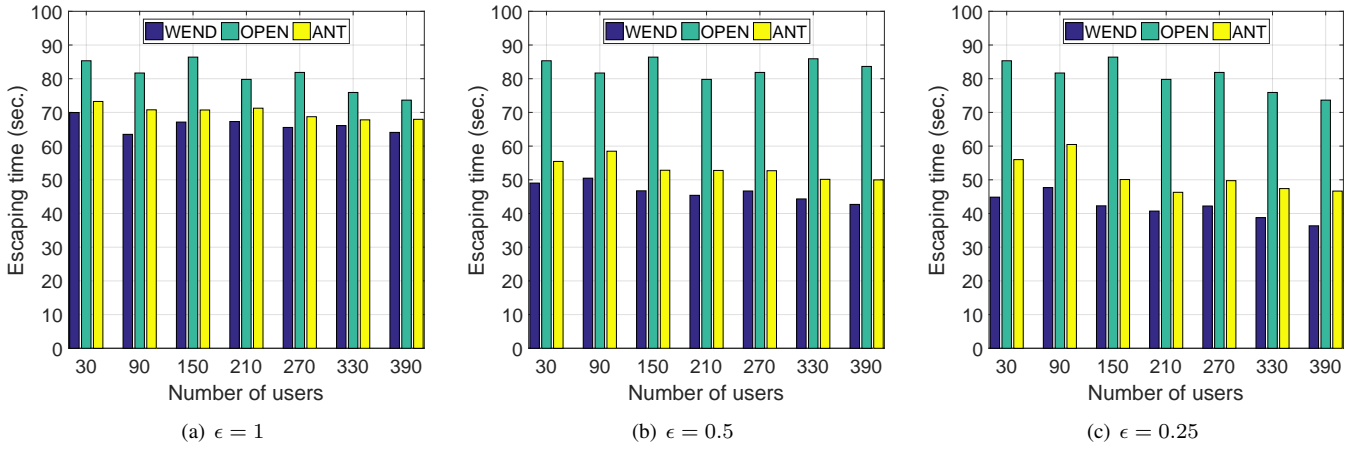


Fig. 8: Comparisons of average escape time for different values of ϵ under different numbers of users

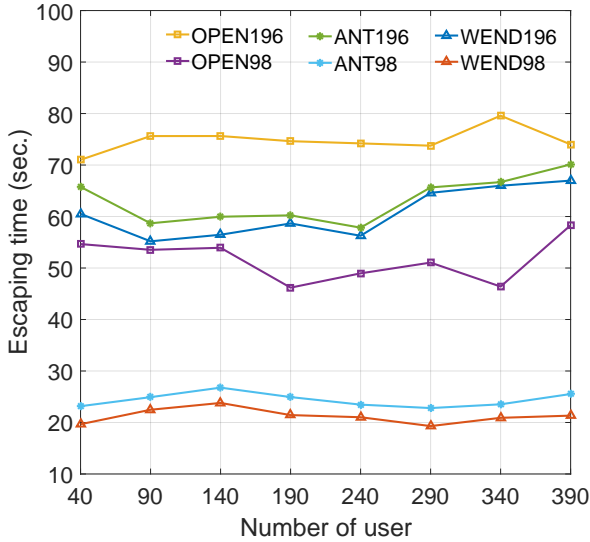


Fig. 9: Comparisons of average escape time for $\epsilon = 0.5$ under different numbers of users and navigation scenarios

ANT the escape time decreases as the deadline for evacuation increases. For OPEN, the escape time is independent of the deadline variation. We also notice when the deadline is set as 60 s, the escape time provided by both WEND and ANT is approximately 65 s which exceeds the specified deadline. The reason is that there is not a route satisfying the worst-case bound when the deadline is set to 60 s. In such a case, the minimum worst-case source-to-destination delay is selected as the escape time by WEND. In addition, we can see that the escaping time of ANT is close to that of WEND when the deadline ranges from 70 to 100. The main reason is the escaping time is less than 10 s, while the typical speed changes every 10 s. That is, there is no dynamic variation when the escaping time is no greater than 10 s.

F. Average emergency response time

An emergency navigation system demands a time-critical response with respect to the provision of navigation decisions.

In this simulation, we compare the navigation efficiency of each method in terms of response time (i.e., the running time of each algorithm). We perform the simulation experiments on a personal computer (Operating System: Windows 10 Education) equipped with Intel Core i5-8400 CPU @ 2.8GHz and 8.0 GB memory. Fig. 11c shows the response time of OPEN, WEND, ANT, and PPA (a pseudo-polynomial algorithm for our problem, which does not round and scale typical delay). We can see that OPEN achieves a similar result in terms of the response time compared with WEND, while the path provided by this scheme cannot guarantee users' escape within the deadline, which extremely jeopardizes the users' chances of survival. We also notice that our approach is much superior to ANT and PPA.

VII. DISCUSSION AND CONCLUSION

Emergency navigation is essential for passengers on a damaged ship. Considering the deadline for ship evacuation and the dynamics of travel time on each segment caused by the changing heel angle, it is challenging to ensure passenger survival. In this study, based on graph theory, we design a Hard-Real-Time emergency navigation strategy in dynamic graphs to minimize escape time, WEND, for ship evacuation. Our method can identify a navigation path with a no more than $(1+\epsilon)d_T^p(\mathcal{T})$ typical delay while guaranteeing the avoidance of dynamic hazardous regions and respecting the specified deadline under all circumstances in scenarios where the typical delay estimations are dynamic. Extensive simulations are used to demonstrate the advantages of our method.

In this work, we do not take into account the physical and psychological factors of moving passengers and the capacity constraints of roads, which is to be addressed in our future work. In addition, the load balancing among different exits is also an important topic requiring an in-depth investigation.

REFERENCES

- [1] E.-G. Ryu, C.-S. Yang, and J.-W. Choi, "Fundamental research for escape guidance system development of passenger ship," *Journal of Korean Navigation and Port Research*, vol. 41, no. 2, pp. 39–46, 2017.

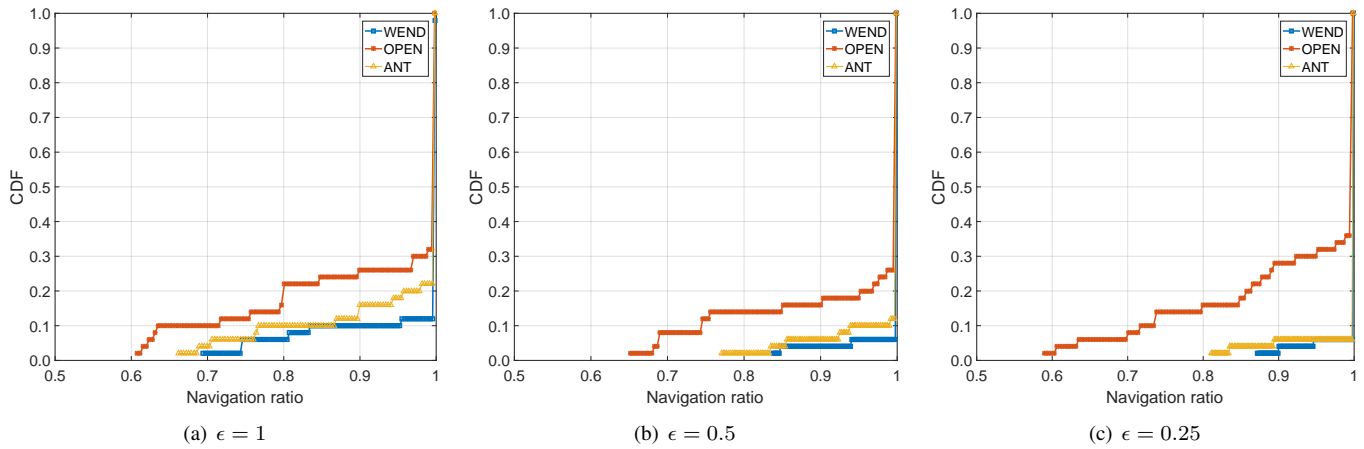


Fig. 10: Comparisons of navigation success ratio for navigation scenario (i) under different values of ϵ

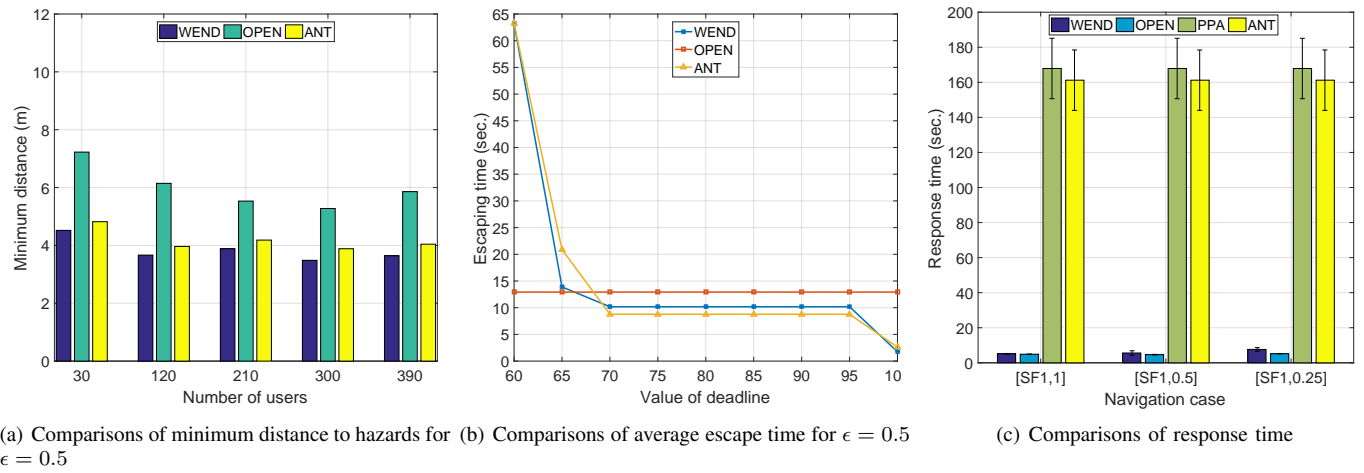
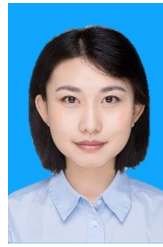


Fig. 11: Comparisons of minimum distance to hazards, average escape time, and the response time

- [2] D. Lee, H. Kim, J.-H. Park, and B.-J. Park, "The current status and future issues in human evacuation from ships," *Safety Science*, vol. 41, no. 10, pp. 861–876, 2003.
- [3] P. Gubian, M. Piccinelli, B. Neri, P. Neri, and F. Giurlanda, "The disaster of costa concordia cruise ship: An accurate reconstruction based on black box and automation system data," in *2015 2nd International Conference on Information and Communication Technologies for Disaster Management (ICT-DM)*. IEEE, 2015, pp. 178–185.
- [4] H. Soliman, "The sinking of the al-salam boccaccio 98 ferry in the red sea: The integration of disaster support system models and emergency management experience," *International Journal of Disaster Risk Reduction*, vol. 4, pp. 44–51, 2013.
- [5] D.-G. Yoon and C.-S. Kim, "Analysis report of the elapse for costa concordia's disaster," *Journal of the Korean Society of Marine Environment & Safety*, vol. 18, no. 4, pp. 331–335, 2012.
- [6] C. Nasso, S. Bertagna, F. Mauro, A. Marinò, and V. Bucci, "Simplified and advanced approaches for evacuation analysis of passenger ships in the early stage of design," *Brodogradnja: Teorija i praksa brodogradnje i pomorske tehnike*, vol. 70, no. 3, pp. 43–59, 2019.
- [7] V. Bucci, A. Marinò, F. Mauro, R. Nabergoj, and C. Nasso, "On advanced ship evacuation analysis," in *22nd International Conference on Engineering Mechanics (IM 2016)*, Svrata, 2016.
- [8] H. Kim, M.-I. Roh, and S. Han, "Passenger evacuation simulation considering the heeling angle change during sinking," *International Journal of Naval Architecture and Ocean Engineering*, vol. 11, no. 1, pp. 329–343, 2019.
- [9] M. Hu, W. Cai, and H. Zhao, "Simulation of passenger evacuation process in cruise ships based on a multi-grid model," *Symmetry*, vol. 11, no. 9, p. 1166, 2019.
- [10] M. Erdelj, M. Król, and E. Natalizio, "Wireless sensor networks and multi-uav systems for natural disaster management," *Computer Networks*, vol. 124, pp. 72–86, 2017.
- [11] M. Abdulkarem, K. Samsudin, F. Z. Rokhani, and M. F. A. Rasid, "Wireless sensor network for structural health monitoring: A contemporary review of technologies, challenges, and future direction," *Structural Health Monitoring*, vol. 19, no. 3, pp. 693–735, 2020.
- [12] C. Wang, H. Lin, R. Zhang, and H. Jiang, "Send: A situation-aware emergency navigation algorithm with sensor networks," *IEEE Transactions on Mobile Computing*, vol. 16, no. 4, pp. 1149–1162, 2016.
- [13] S. M. Kumar and L. Lakshmanan, "A situation emergency building navigation disaster system using wireless sensor networks," in *2018 International Conference on Communication and Signal Processing (ICCCSP)*. IEEE, 2018, pp. 0378–0382.
- [14] L. Wang, Y. He, W. Liu, N. Jing, J. Wang, and Y. Liu, "On oscillation-free emergency navigation via wireless sensor networks," *IEEE Transactions on Mobile Computing*, vol. 14, no. 10, pp. 2086–2100, 2014.
- [15] C. Wang, H. Lin, and H. Jiang, "Cans: Towards congestion-adaptive and small stretch emergency navigation with wireless sensor networks," *IEEE Transactions on Mobile Computing*, vol. 15, no. 5, pp. 1077–1089, 2015.
- [16] A. Gunathillake and A. V. Savkin, "Mobile robot navigation for emergency source seeking using sensor network topology maps," in *2017 36th Chinese Control Conference (CCC)*. IEEE, 2017, pp. 6027–6030.
- [17] I. Bačkalov, G. Bulian, J. Cichowicz, E. Eliopoulou, D. Konovessis, J.-F. Leguen, A. Rosén, and N. Themelis, "Ship stability, dynamics and safety: Status and perspectives from a review of recent stab conferences and issw events," *Ocean Engineering*, vol. 116, pp. 312–349, 2016.
- [18] K. J. Rawson and E. C. Tupper, *Basic Ship Theory Volume 1*. Butterworth-Heinemann, 2001, vol. 1.
- [19] H. Nowacki, "Archimedes and ship design," in *Archimedes in the 21st*

- Century. Springer, 2017, pp. 77–112.
- [20] Y. Han and H. Liu, “Modified social force model based on information transmission toward crowd evacuation simulation,” *Physica A: Statistical Mechanics and its Applications*, vol. 469, pp. 499–509, 2017.
 - [21] R. Zhou, Y. Cui, Y. Wang, and J. Jiang, “A modified social force model with different categories of pedestrians for subway station evacuation,” *Tunnelling and Underground Space Technology*, vol. 110, p. 103837, 2021.
 - [22] L.-W. Chen and J.-X. Liu, “Time-efficient indoor navigation and evacuation with fastest path planning based on internet of things technologies,” *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 51, no. 5, pp. 3125–3135, 2019.
 - [23] T. Meyer-König, P. Valanto, and D. Povel, “Implementing ship motion in aeneas—model development and first results,” in *Pedestrian and evacuation dynamics 2005*. Springer, 2007, pp. 429–441.
 - [24] K. Kostas, A.-A. Ginnis, C. Politis, and P. Kaklis, “Motions effect for crowd modeling aboard ships,” in *Pedestrian and Evacuation Dynamics 2012*. Springer, 2014, pp. 825–833.
 - [25] D. Lee, J.-H. Park, and H. Kim, “A study on experiment of human behavior for evacuation simulation,” *Ocean Engineering*, vol. 31, no. 8-9, pp. 931–941, 2004.
 - [26] H. Kim, J.-H. Park, D. Lee, and Y.-s. Yang, “Establishing the methodologies for human evacuation simulation in marine accidents,” *Computers & Industrial Engineering*, vol. 46, no. 4, pp. 725–740, 2004.
 - [27] M. Li, Y. Liu, J. Wang, and Z. Yang, “Sensor network navigation without locations,” in *IEEE INFOCOM 2009*. IEEE, 2009, pp. 2419–2427.
 - [28] C. Buragohain, D. Agrawal, and S. Suri, “Distributed navigation algorithms for sensor networks,” *arXiv preprint cs/0512060*, 2005.
 - [29] Q. Li, M. De Rosa, and D. Rus, “Distributed algorithms for guiding navigation across a sensor network,” in *Proceedings of the 9th annual international conference on Mobile computing and networking*, 2003, pp. 313–325.
 - [30] H. Li and A. V. Savkin, “An algorithm for safe navigation of mobile robots by a sensor network in dynamic cluttered industrial environments,” *Robotics and Computer-Integrated Manufacturing*, vol. 54, pp. 65–82, 2018.
 - [31] S. Allali and M. Benchaiba, “Safe route guidance of rescue robots and agents based on hazard areas dissemination,” in *2017 International Conference on Mathematics and Information Technology (ICMIT)*. IEEE, 2017, pp. 29–37.
 - [32] D. H. Lorenz and D. Raz, “A simple efficient approximation scheme for the restricted shortest path problem,” *Operations Research Letters*, vol. 28, no. 5, pp. 213–219, 2001.
 - [33] R. Hassin, “Approximation schemes for the restricted shortest path problem,” *Mathematics of Operations research*, vol. 17, no. 1, pp. 36–42, 1992.
 - [34] Y. Ma, K. Liu, M. Chen, J. Ma, X. Zeng, K. Wang, and C. Liu, “Ant: Deadline-aware adaptive emergency navigation strategy for dynamic hazardous ship evacuation with wireless sensor networks,” *IEEE Access*, vol. 8, pp. 135 758–135 769, 2020.
 - [35] P. Valanto, “Time-dependent survival probability of a damaged passenger ship ii-evacuation in seaway and capsizing,” *HSVA Report*, no. 1661, pp. 42–62, 2006.
 - [36] R. A. Klein, “Cruises and bruises: safety, security and social issues on polar cruises,” *Cruise tourism in polar regions: Promoting environmental and social sustainability*, pp. 57–74, 2010.
 - [37] P. Boguslawski, L. Mahdjoubi, V. Zverovich, and F. Fadli, “Automated construction of variable density navigable networks in a 3d indoor environment for emergency response,” *Automation in Construction*, vol. 72, pp. 115–128, 2016.
 - [38] Y. Qiao, D. Han, J. Shen, and G. Wang, “A study on the route selection problem for ship evacuation,” in *2014 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. IEEE, 2014, pp. 1958–1962.
 - [39] W.-J. Na, B.-H. Son, and W.-H. Hong, “Analysis of walking-speed of cruise ship passenger for effective evacuation in emergency,” *Medico-Legal Update*, vol. 19, no. 2, 2019.
 - [40] P. Guo, J. Xia, M. Zhou, R. A. Falconer, Q. Chen, and X. Zhang, “Selection of optimal escape routes in a flood-prone area based on 2d hydrodynamic modelling,” *Journal of Hydroinformatics*, vol. 20, no. 6, pp. 1310–1322, 2018.
 - [41] H. Ortega-Arranz, D. R. Llanos, and A. Gonzalez-Escribano, “The shortest-path problem: Analysis and comparison of methods,” *Synthesis Lectures on Theoretical Computer Science*, vol. 1, no. 1, pp. 1–87, 2014.



Yuting Ma is currently pursuing the Ph.D. degree in marine navigation from the Wuhan University of Technology (WUT), Wuhan, China. Her research interests focus on emergency navigation and evacuation.



Kezhong Liu received the B.S. and M.S. degrees in marine navigation from the Wuhan University of Technology (WUT), Wuhan, China, in 1998 and 2001, respectively. He received the Ph.D. degree in communication and information engineering from the Huazhong University of Science and Technology, Wuhan, China, in 2006. He is currently a professor with School of Navigation, WUT. His active research interests include indoor localization technology and data mining for ship navigation.



Mozi Chen received the B.S. degree in electric engineering from the Hubei University of Technology, China, in 2013, the M.S. degree in navigation engineering from the Wuhan University of Technology (WUT), China, in 2016, and the Ph.D. degree from the WUT, China, in 2020. He is currently an associate researcher in WUT. His research work has been focusing on wireless sensing techniques and machine learning algorithms for human localization, emergency navigation and activity recognition in mobile environment, i.e., cruise ships.



less network resource management, and embedded operating systems.

kehao Wang received the B.S. degree in Electrical Engineering, M.S. degree in Communication and Information System from Wuhan University of Technology, Wuhan, China, in 2003 and 2006, respectively, and Ph.D. degree in the Department of Computer Science, the University of Paris-Sud XI, Orsay, France, in 2012. In 2013, he joined the School of Information Engineering at the Wuhan University of Technology, where he is currently an associate professor. His research interests include cognitive radio networks, wireless network resource management, and embedded operating systems.



Kai Zheng received the Ph.D. degree with the School of Geodesy and Geomatics, Wuhan University. He is currently Associate Researcher with the Wuhan University of Technology. His research interests are GNSS precise positioning techniques.



Xuming Zeng received the Ph.D. degree in communication engineering from the China University of Geosciences, Wuhan, China, in 2018. He is currently working toward the postdoctor with the Traffic and Transportation Engineering, School of Navigation, Wuhan University of Technology, Wuhan, China, where he has been studying wireless ad-hoc network for shipboard environment since 2018. From 2016 to 2017, he was a joint training Ph.D. student with Electrical and Computer Engineering, College of Engineering, Florida State University, Tallahassee, FL, USA. His research interests include routing protocols, MAC, QoS, clustering, radio resource management, traffic engineering, and performance analysis, for both wired and wireless networks.