Hindawi
Journal of Advanced Transportation
Volume 2018, Article ID 4295419, 13 pages
https://doi.org/10.1155/2018/4295419

Research Article

Dynamic Route Network Planning Problem for Emergency Evacuation in Restricted-Space Scenarios

Yi Hong,[1] **Deying Li**,[2] **Qiang Wu**,[3] **and Hua Xu**[1]

[1]*Information Engineering College, Beijing Institute of Petrochemical Technology, Beijing 102617, China*
[2]*School of Information, Renmin University of China, Beijing 100872, China*
[3]*National Engineering Research Center of Coal Mine Water Hazard Controlling, China University of Mining and Technology, Beijing, Beijing 100083, China*

Correspondence should be addressed to Yi Hong; hongyi@bipt.edu.cn

Received 22 March 2018; Revised 14 May 2018; Accepted 17 May 2018; Published 27 June 2018

Academic Editor: Zhi-Chun Li

We investigate a dynamic route planning problem in restricted-space evacuation, namely, the *Multiobjective Dynamic Route Network Planning (MODRNP) problem*. It models the multisource to multidestination evacuation in restricted-space scenarios, with the objectives of minimizing the whole evacuation delay and maximizing the evacuation efficiency. We study the problem in 3D scenarios, which can provide intuition vision for the geographic space and contribute to the evacuation plan and implementation. Based on the auxiliary graph transformation, we propose a heuristic algorithm referred to the classical problem, Minimum Weighted Set Cover. We finally conduct extensive experiments to evaluate the performance of the proposed algorithm and give an application instance on a typical kind of restricted-space scenarios. The results indicate that the proposed algorithm outperforms the existing alternatives in terms of the utilization as well as timeliness.

1. Introduction

As a significant research problem, disaster prevention and relief became one of the new requirements for social development in recent years, and emergency evacuation planning is an important part of rescue and relief work. Emergency evacuation generally contains the evacuee evacuation [1] and the relief resource deployment [2], and the evacuee evacuation is the most urgent work of disaster relief. An effective evacuation path planning strategy can not only effectively reduce the evacuation time consumption, but also prevent unnecessary congestion in the evacuation process, even secondary damage. Thus, the optimization design and theoretical analysis for evacuation path planning are of great practical significance [3].

According to the disaster scenarios, evacuation path planning can be classified into free-space (e.g., urban streets and squares) planning and restricted-space (e.g., underground pipeline, tunnel, and mine laneway) planning [4, 5]. The former one has most applications in city-security and antiterrorist, which has been well studied based on one-source to one-destination model or one-source to multidestination model.

The later one involves in a large range of industrial applications, i.e., the evacuation planning in fire disaster, gas explosion, or water inrush in underground mine. The restricted-space has several complex characteristics which are different from the free space: **(a) the space and time consuming limitations on entrance and evacuation. The density of personnel is sparser; (b) the unsuitability of long operating time. The reliability of temporary shelters is weaker; (c) the trapped-vulnerability of internal structure in disasters.** When a disaster occurs, the number of available evacuation paths is much less. Thus, the restricted-space has higher accident rate and brings more difficulties and challenges to rescue and relief work, which has been urgent to be studied. To improve the quality of emergency response in restricted-space planning, we focus on dynamic path planning based on the multisource to multidestination model.

In practical, restricted-space emergency evacuations have different evacuation requirements according to the accident type, nature, and distribution. On the one hand, considering the limitations and unsuitability of restricted spaces, it may cause local or global congestions when evacuation traffic load is too large, e.g., the evacuees intensively rushing to a certain exit. Thus, evacuation requirements do not only include the shortest evacuation time consumption, but also the unobstructed connectivity of route network. The goals of the restricted-space emergency evacuation are minimizing the global evacuation time consumption and maximizing the utilization rate of each exit (balancing among the exits' traffic loads). **To minimize the global evacuation time consumption**, each evacuee tends to choose the shortest path and nearest exit, which may cause the imbalance of exit utilization, i.e., most evacuees intensively rushing to a certain exit via their shortest paths. **To maximize the utilization rate of each exit**, the evacuees should dispersedly choose the exits, which may cause that some evacuee chooses the suboptimal path rather than the shortest one and its evacuation time consumption will be lengthened. Thus, the two objectives are conflicting and they need a tradeoff to balance their realization, i.e., each evacuee should choose the global optimum rather than local optimum. On the other hand, due to the trapped-vulnerability of restricted spaces, the static planning strategies cannot adapt to the dynamic spreading of disasters or accidents and other environment changes. Thus, it is necessary to design dynamic evacuation planning based on environment changes. Based on the above considerations, we propose new evacuation routes planning problem, the Multiobjective Dynamic Route Network Planning (MODRNP) problem, and aim to design dynamic route network planning strategies and give theoretical analysis.

In this paper, firstly we introduce the restricted-space model and formulate the Multiobjective Dynamic Route Network Planning (MODRNP) problem in restricted-space evacuation. Secondly, we propose dynamic and global-optimized strategies to construct the evacuation route network, whose performance will be evaluated by the simulations and instance experiments at last. The rest of the paper is organized as follows. We summarize the related work in Section 2. Section 3 introduces the restricted-space model and problem definitions. Section 4 introduces the strategy framework for MODRNP Problem and describes the algorithms. Simulation results and corresponding discussions are given in Section 5. Section 6 concludes this paper.

2. Related Works

As a crucial kind of combinational optimization problem, the path planning problem has been widely studied such like recent research works on traveling salesman problem (TSP)[6], traffic assignment/vehicle routing problem [7–9], and evolutionary optimization problems, e.g., general transportation planning problems, facility location problem, and roadway repair problem [10]. Based on the existing theoretical achievements, the research on evacuation path planning can be classified as static planning and dynamic planning.

Research on the **static path planning problem provides theoretical guideline for the emergency analysis and escaping exercise preparation.** Most algorithms for it are based on Dijkstra algorithm, improved Dijkstra algorithm, Floyd algorithm, the first k shortest path algorithm, A^* algorithm, and dynamic programming. Under the static problem model, the solutions generally simplified the disaster type, evacuee psychological pattern, and other factors by introducing the path equivalent length, with the assumption that the disaster would be spreading steadily.

The studies on the **dynamic path planning problem** are the basis of real-time decision making in emergency response and management in disaster/accident. The solutions can be classified into two kinds, the approximate algorithms with high theoretical precision and the heuristic ones with high efficiency.

To design approximate algorithms for the dynamic problem model, most researchers firstly transformed the problem into the classical network flow problem model [11–14] and then utilized the classical polynomial time algorithms to design the path selection strategy. **By applying the maximum flow algorithm**, Dunn and Newton [15] constructed a path network with the goal of guiding the most survivors in accordance with the constructed path network under the condition of permission capacity. **By applying the minimum cost flow algorithm**, Yamada [16] assigned evacuation traffic aimed at minimizing the overall length of the chosen evacuation paths, and Cova and Johnson[17] focused on complex path network with the multiple objectives of minimizing the intersections between the evacuation paths and minimizing the overall evacuation length. **By applying the fast flow algorithm**, the authors in [18, 19] aimed at the application scenarios modeled as directed graph. But the above algorithms could be only applied to the case of grid structure path network and unified path capacity. **By applying the contraflow algorithm**, the authors in [20] considered the setup time of the contraflow operation and proposed a two-layer algorithm to minimize the evacuation time consumption and the setup time for flow patterns. Since the algorithms for the flow problems have high time or space complexity, most of these strategies lack high efficiency in practice.

To design heuristic algorithms for the dynamic problem model, researchers in [21, 22] designed simulated evolutionary algorithms. They took full advantages of the positive feedback information mechanism of ant colony algorithm and the fast convergence of the genetic algorithm [23]. For the airport emergency evacuation scenarios, the authors in [24] performed a series of simulations based on an agent-based model (ABM) to determine the collective behaviors and the overall evacuation time, which are affected by the environment and complex structures. And [25] was motivated by the improvement of the marine evacuation system plan in a practical case, Dalian offshore airport of China, and developed a modeling framework to evaluate and optimize the emergency evacuation capability. For the crowded building emergency evacuation scenarios, the authors in [26] studied the human behavior of crowd evacuation to guide the design of buildings and the reinforcement of security management. The proposed algorithms can solve the path planning problem in densely populated scenarios. Due to the sparsity of personnel in restricted spaces, these strategies cannot be applied directly in the evacuation planning.

The above related work had limitations at three aspects. Firstly, most research works considered application scenarios for two-dimensional (2D) planes, in which the proposed algorithms cannot adapt to three-dimensional (3D) application scenarios. Secondly, most existing solutions for the path network construction problem were with high time or space complexity, which may bring low efficiency in restrict space evacuation. Thirdly, the existing researches on dynamic planning of emergency evacuation paths were still limited; e.g., the authors in [27] conducted research on the vehicle routing problem for postdisaster scenarios. And their application effect in practical is not ideal in terms of dynamic adjustment and timely feedback. Therefore, we intend to introduce a new dynamic planning problem model and utilized new technology in solution for restricted-space scenarios.

3. Problem Formulation

3.1. Restricted-Space Model. We focus on the emergency evacuation in confined space works and model such a densely populated restricted-space, as a 3D connected graph $G = (V, E)$, where $V = \{v_1, \cdots, v_n\}$ is the set of predetermined observation vertices in the space, E is the set of bidirectional edges, and for each $v_i \in V$, $v_i = (ID_i, x_i, y_i, z_i)$ and for each $e_k \in E$, $e_k =< ID_i, ID_j > (1 \le i < j \le n)$. For any vertex subset $V' \subseteq V$, $G[V']$ is the subgraph of G induced by V'. Similarly, $G[E']$ is the subgraph of G induced by an edge subset $E' \subseteq E$. The source set S and the destination set D are important subsets of the vertex set in G, which are composed of the evacuees' initial positions and the feasible exits, respectively. Here we assume S and D are predefined.

When a disaster or an accident happens, the evacuees need to escape via the edge/segment in the restricted-space in the shortest possible time. And the traffic capacity of each edge is affected by several influence factors, which need to be considered in the decision of the most reliable evacuation paths. We list the main influence factors of the restricted-space as follows.

(a) *Edge types:* $\beta_1(e_k)$.The properties like the shape and slope of the edge/section decide the different types of the edge. Different edge types have different level effects on evacuee evacuation. Here we consider the scope of the edge, and for any pair of edges $< ID_i, ID_j >$ and $< ID_j, ID_i >$, $\beta_1(< ID_i, ID_j >) = -\beta_1(< ID_j, ID_i >) \ (1 \le i < j \le n)$.

(b) *Passibility:* $\beta_2(e_k)$. The passibility of the edge is greatly influenced by the obstacles and the vehicle of the evacuees. We assume that the evacuees adopt a uniform activity manner and have the same traveling speed. For any pair of edges $< ID_i, ID_j >$ and $< ID_j, ID_i >$, $\beta_2(< ID_i, ID_j >) = \beta_2(< ID_j, ID_i >) \ (1 \le i < j \le n)$.

(c) *Exit escaping priority:* $priority(d_i)$. Due to the exits' security levels, structural characteristics, or functional characteristics, different exits have different escaping priorities $\{priority(d_i) \mid \forall d_i \in D\}$ which have influence on the edge's traffic.

(d) *Evacuation priority:* $\alpha(e_k)$. The factor is generated from the incident relationship with the destinations in D, i.e., if edge e_k is an incident edge with more than one d_i in D, $\alpha(e_k) = \max_{d_i \in D \text{ and } d_i \in e_k} priority(d_i)$. For any pair of edges $< ID_i, ID_j >$ and $< ID_j, ID_i >$, $\alpha(< ID_i, ID_j >) = \alpha(< ID_j, ID_i >) \ (1 \le i < j \le n)$.

(e) *Travel length:* $length(e_k)$. The geometric length of the edge. For any pair of edges $< ID_i, ID_j >$ and $< ID_j, ID_i >$, $length(< ID_i, ID_j >) = length(< ID_j, ID_i >) \ (1 \le i < j \le n)$.

Based on these influence factors, the traffic capacity on an edge can be expressed in terms of the equivalent length/edge weight: $weight(e_k) = length(e_k) \cdot \beta_1(e_k) \cdot \beta_2(e_k) \cdot \alpha(e_k), 1 \le k \le |E|$. Thus, the constrained space can be further modeled as an edge-weighted graph $G = (V, E, W)$. Here we adopt travel length instead of traveling time to be the basis of traffic capacity for the reason that the traveling speed of each evacuee is regarded as a uniform value.

3.2. Disaster Spreading Model. When the disaster or accident happens, the spreading speed and mode have great influence on the decision of evacuation route. Generally, the disaster spreading pattern is determined by several influence elements, e.g., the fire spreading pattern is effected by carbon monoxide in the volume flow of smoke and fire gases CO, oxygen in the fire place O_2, heat flow, and so on. And the spreading pattern is directly reflected in the speed of disaster on each edge. Generally the spreading pattern can be modeled as a statistical model depending on the disaster itself.

To this end, we value the maximum spreading time duration of the disaster as T_{dis}; i.e., it takes T from the disasters happening to its whole coverage. Note that we assume that all the evacuees can find the feasible evacuation path and their evacuation time consumption is not exceeding T_{dis}. We also assume that T_{dis} can be splitted as several timeslots according to the spreading rule and the evacuation time can be measured as the same timeslot. Thus, the disaster spreading pattern can be modeled as a function of $\mathcal{F}(G_t, t)$, where $G_t = (V_t, E_t, W_t)$ is the temporal space subgraph of G determined by disaster spreading speed in the t-th timeslot. And the influence factor of the disaster is as follows.

(f) *Disaster spreading status:* $\beta_3(e_k, t)$. In each timeslot, we value $\beta_3(e_k, t)$ according to $\mathcal{F}(G_t, t)$. For any pair of edges $< ID_i, ID_j >$ and $< ID_j, ID_i >$, $\beta_3(< ID_i, ID_j >, t) = \beta_3(< ID_j, ID_i >, t) \ (1 \le i < j \le n)$.

Based on the influence factor, the edge weight in G_t can be further expressed as $weight(e_k, t) = weight(e_k) \cdot \beta_3(e_k, t), 1 \le k \le |E|$. Note that the edge weight is a time-related measure only in G_t depending on disaster spreading status.

3.3. Problem Definitions. We consider a Multiobjective Dynamic Route Network Planning problem in a restricted-space evacuation, which is to design a staged and adjusted construction scheme of evacuation path network from all the evacuees in S to the emergency exit set D in a 3D space graph $G = (V, E, W)$.

Firstly, the problem is motivated from two points of view: the one is for each evacuee and the goal is to decide its globally ideal escape exit based on evacuation priority and find the escaping path to the ideal exit; the other one is for each escape exit. In most of practical applications, the feasible escape exits have different evacuation priorities; e.g., in underground coal mine evacuation, the auxiliary and main shafts have higher priority than chambers or mobile capsules. Thus, to enhance the utilization rate of each exit, it is needed to assign relatively balanced count of evacuees to the exit such that the case of congesting in some an overdrawn exit can be avoided.

Secondly, the problem has two objectives; one is in time dimension, i.e., minimizing the whole evacuation travel length. Here *the whole evacuation travel length* is defined as the travel length consumed by the latest evacuee, which can be referred to as the length of the longest escaping path, $\max_{\forall d_i \in D, \forall s_j \in S} length(d_i, s_j)$. And the other one is in space dimension, i.e., maximizing the priority-oriented evacuation efficiency. *The priority-oriented evacuation efficiency* is a global measure representing the balance performance among each exit's utilization efficiency; i.e., a smaller efficiency stands for a higher balance performance. And the priority-oriented evacuation efficiency is defined as the maximum value among the utilization efficiency differences between any pairs of exits, $\max_{d_p, d_q \in D} |priority(d_p)/|d_p| - priority(d_q)/|d_q||$, where d_i represents two meanings, the exit d_i itself and the set of the sources in S escaping from d_i finally.

Thirdly, the problem has dynamic constrains, i.e., disaster spreading status, which is the most influential one among the factors for traffic capacity and is varying along with the evacuee evacuation. In each timeslot, the disaster spreading pattern is regarded as an input item of the path planning process and it can be modeled as a function of $\mathcal{F}(G_t, t)$, where G_t is the temporal version of G.

Our problem in restricted-space evacuation is defined as follows, which is a staged problem model.

Definition 1 (Multiobjective Dynamic Route Network Planning (MODRNP) problem).

Input: Given a 3D restricted-space model, graph $G = (V, E, W)$, source set S, and destination set D with its priority set, $priority(D)$, the disaster spreading pattern $\mathcal{F}(G_t, t)$.

Find: Find the route network in each timeslot t from all the sources to their optimal destinations, and the whole route network is constructed with two goals, minimizing the whole evacuation travel length $\max_{\forall d_i \in D, \forall s_j \in S} length(d_i, s_j)$ and maximizing the priority-oriented evacuation efficiency $\max_{d_p, d_q \in D} |priority(d_p)/|d_p| - priority(d_q)/|d_q||$.

Theorem 2. *MODRNP Problem in restricted-space evacuation is NP-hard.*

Proof. In order to proof the hardness of MODRNP Problem, we firstly decompose the problem in each timeslot as a subproblem and rewrite the subproblem into a mathematical formulation as below: In a 3D connected and edge-weighted graph $G_t = (V_t, E_t, W_t)$, given two vertex subsets of V_t, the source set S_t and the destination set D, and D's priority set of $priority(D)$, the problem is to find a path network from S_t to D such that

(i) the maximum difference among D's utilization efficiencies, $\max_{d_p, d_q \in D} |priority(d_p)/|d_p| - priority(d_q)/|d_q||$, can be minimized, where d_p is not only a destination in D, but also a set of nodes in S_t and connected to d_p by the constructed paths,

(ii) the maximum path length among all sources and destinations, $\max_{\forall d_i \in D, \forall s_j \in S_t} length(d_i, s_j)$, can be minimized.

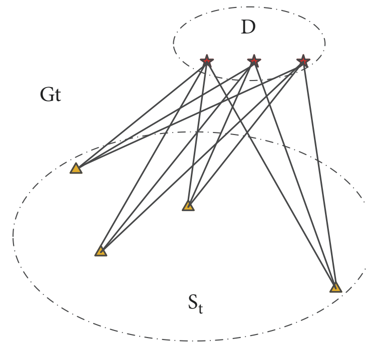


Figure 1: Bipartite graph transformation.

Then we consider a special case of it: Case (a). $\forall d_i \in D, \forall s_i \in S_t$, $length(d_i, s_j)$s are a uniform value; i.e., the maximum value in (ii) is unified as a constant. Thus, the minimization of the maximum value can be regarded a default implement. Based on Case (a), we transform these isometric paths into the corresponding one-hop edges; i.e., G_t is transformed into a bipartite graph with the two vertex sets S_t and D, $G_t[D, S_t]$ as shown in Figure 1.

Case (b). $\exists d_0 \in D$, $priority(d_0) = 0$, i.e., the objective measure in (i) $\min \max_{d_p, d_q \in D} |priority(d_p)/|d_p| - priority(d_q)/|d_q|| = \min_{d_p \in D} |priority(d_p)/|d_p| - 0/|d_0|| = \min_{d_p \in D}(priority(d_p)/|d_p|)$. For each node $d_i \in D$, d_i can be used to represent its accessible nodes in S_t, i.e., $d_i \subseteq S_t$. When each d_p is assigned the weight of $(priority(d_p) \cdot |S_t|)/(|d_p| \cdot |D|)$, the problem becomes to find a conditional set cover in the bipartite graph $G_t[D, S_t]$.

Based on the above problem variant, the subproblem in Cases (a) and (b) is to find a subcollection $\mathscr{C} \subseteq \mathscr{D}$ (here $\mathscr{D} = \{d_1, d_2, \ldots, d_{|D|}\}$) such that $\bigcup_{d_i \in \mathscr{C}} d_i = S$ and the total weight of d_i in $\mathscr{C}$ is minimized. Here we review an important and classical NP-hard problem, **Minimum Weighted Set Cover (MWSC) Problem**: Given a set $\mathscr{X}$ composed of n elements, a collection $\mathscr{D}$ of m subsets of $\mathscr{X}$ ($\mathscr{D} = \{\mathscr{X}_1, \mathscr{X}_2, \ldots, \mathscr{X}_m\}$), and each $\mathscr{X}_i \subseteq \mathscr{X}$ $(1 \leq i \leq m)$ has a weight $w(\mathscr{X}_i)$, the problem is to find a subcollection $\mathscr{C} \subseteq \mathscr{D}$ such that $\bigcup_{\mathscr{X}_i \in \mathscr{C}} \mathscr{X}_i = \mathscr{X}$ and $\sum_{\mathscr{X}_i \in \mathscr{C}} w(\mathscr{X}_i)$ is minimized. It is easy to discover that this special version of each subproblem is equivalent to MWSC problem, which was proved to be NP-hard [28]. Therefore, MODRNP Problem is NP-hard in general. □

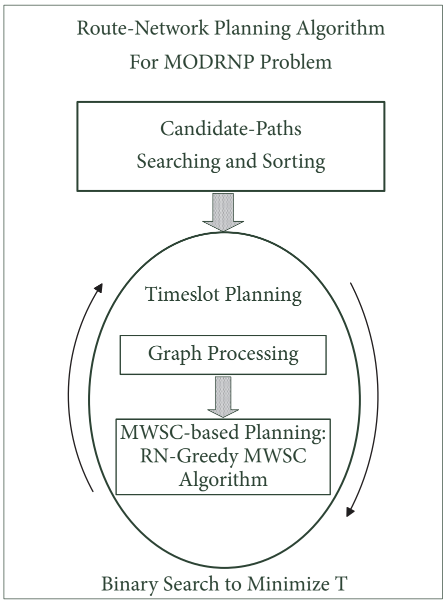


FIGURE 2: The framework of Route Network Planning Algorithm.

4. Multiobjective Dynamic Route Network Planning Algorithm

To solve the NP-hard problem MODRNP, we propose a partition strategy (the framework is shown in Figure 2), which has a uniform algorithm and progressive inputs in each timeslot. The division of timeslot is determined by the disaster spreading function $\mathscr{F}(G_t, t)$ and it generates the temporal graph G_t in timeslot t. In each timeslot, we construct the route network according to the following iterations.

In each timeslot t,

(i) Phase 1: update the space graph according to the spreading pattern, $G_t \leftarrow \mathscr{F}(G_t, t)$, where G_t can be regarded as a progressive input for the algorithm.

(ii) Phase 2: search for the current route network $\mathscr{P}_t$ in G_t, which connects the current source set S_t to the set D. Since the evacuees from the current sources may not reach the destinations D in this timeslot limited by the evacuee speed, it is also needed to extract the path set $\mathscr{P}_t^*$ from S_t to their destinations in the current timeslot, collected by D_t.

(iii) Phase 3: update S_t to D_t as the new source set in the next timeslot and enter timeslot $t + 1$.

In Phase 2, the searching strategy consists of three steps.

Firstly, we find all the shortest paths from each node of S_t to each node of D in G_t based on Dijkstra Algorithm and record all the obtained paths as a subgraph G_t'.

Secondly, we sort all the paths of G_t' in the nondecreasing order of their lengths to obtain an ordered path set $G_t' = \{path_1, path_2, \cdots, path_q\}$ as shown in step (3) in Algorithm 1.

Thirdly, we construct a route network from the current source set S_t to D, whose details are as follows: (1) construct a subgraph G_t' composed of all the shortest paths from each node of S_t to the whole destinations of D via Dijkstra Algorithm; (2) design a greedy strategy to select the optimal whole destination for each node of S_t in G_t' based on Minimum Weighted Set Cover and record the selected shortest paths as RN; (3) for each node in S_t, decide its current destination on its path in G_t' and store the corresponding paths into RN. The construction is under the goals of minimizing the longest path and maximizing the priority-oriented evacuation efficiency.

The former objective is realized by a binary search as shown by the while loop in steps (5)-(17). In each iteration of the binary search, for each $path_i \in G_t'$, starting from $Mid = \lfloor (1 + q)/2 \rfloor$, we **temporarily** remove all the $path \in G_t'$ whose length satisfies $length(path) \geq length(path_{Mid})$ and check if the subgraph of G_t' can still construct a route network without those edges, which is verified by a searching algorithm. **If a route network can be constructed, we obtain a route network RN on the subgraph of G_t' and proceed with the increased Mid. Otherwise, we keep those edges in G_t' and proceed with the decreased Mid.**

The latter objective, balancing the evacuation efficiency among all the nodes of D, is implemented by the route network construction in each iteration of the binary search: we propose a path planning algorithm, **RN-Greedy-MWSC**(S_t, D) (Algorithm 2). To compute evacuation paths from multiple sources to their most suitable destinations, we referred to the classical approximation algorithm with the ratio of $\ln(n + 1)$ for MWSC problem: select a subset $\mathscr{T} \in \mathscr{F}$ that minimizes the whole redundant weight $(w(\mathscr{T})/|\mathscr{T} \bigcap \mathscr{U}|)$ in each loop, where $\mathscr{U}$ stands for the uncovered elements in $\mathscr{X}$; repeat the greedy selection until the set of $\mathscr{T}$s cover $\mathscr{X}$. Thus, the criteria of greedy selection here are the redundant utilization, which is defined in step (8) of Algorithm 2; i.e., we repeat selecting a best d_i with the minimum $|redutil(d_i)/|d_i \bigcap \mathscr{S}||$ value in each iteration until each node S_t can be reached by the selected d_is.

Theorem 3. *The time complexity of Route Network Planning Algorithm for MODRNP Problem is* $\mathbf{O}(\mathbf{n}^2)$.

Proof. For the four phases for Algorithm 1, the first one (Step (2)) is constructing all the candidate paths between S_t and D which has $\mathbf{O}(\mathbf{n}^2)$ time complexity. In the second phase (Step (3)), sorting the candidate paths based on their lengths takes $\mathbf{O}((|\mathbf{S}| * |\mathbf{D}|)^2)$ time. Thus, the time cost in this phase is $O(n \log n)$. For Phase 3 (Steps (5)-(17)), the binary search for the current route network construction of the auxiliary graph has the time complexity of $\mathbf{O}(|\mathbf{D}| \cdot \log(|\mathbf{S}| \cdot |\mathbf{D}|))$. In the final phase (Steps (18)-(21)), the worst-case time complexity is $\mathbf{O}(|\mathbf{S}|)$. To conclude, the whole algorithm has the time complexity of $\mathbf{O}(\mathbf{n}^2)$. □

5. Performance Evaluation

5.1. Simulation Plan. We perform simulation experiments based on the random generation of 3D networks $G = (V, E)$. The vertex set V stands for the observation nodes which are randomly deployed for a constrained space, and the deployment of it is randomly assigning n nodes' 3D

Construct a route network from S_t to D_t in timeslot t. *in:* G_t, S_t, D *out:* $\mathcal{P}_t^*, D_t$
Candidate-paths searching: Find all the shortest path from each node of S_t to each node of D in G_t based on Dijkstra Algorithm, and record all the paths into subgraph G_t'. Note that for each $d_i \in D$, d_i is also represented as the set of the sources in S_t which can reach it in G_t.
Candidate-paths sorting: Sort the paths in G_t' according to their weights in the non-decreasing order and store the order in $G_t' = \{path_1, path_2, \cdots, path_q\}$.
Set $\mathcal{P}_t^* \longleftarrow \emptyset$, $D_t \longleftarrow \emptyset$, $Min = 1$, $Max = q$, $Mid = 0$
while $(Min \le Max)$ **do**
  $Mid = \lfloor \frac{Min + Max}{2} \rfloor$, $G_{temp} = \{path \mid path \in G_t'$ and $length(path) \ge length(path_{Mid})\}$;
  $\forall d_i \in G_t'$, $d_i \longleftarrow d_i \setminus \{s_j \in G_{temp}\}$;
  Graph processing: $G_t' \longleftarrow G_t' \setminus G_{temp}$, $D \longleftarrow D \setminus \{d_i \mid \forall d_i \in G_{temp}$ and $|d_i| = 0\}$;
  RN-Greedy-MWSC(S_t, D)**:** Find whether a path set can be constructed in G_t';
  if (RN-Greedy-MWSC(S_t, D) returns **False**) **then**
    $Max = Mid$;
    $\forall d_i \in G_{temp}$, $d_i \longleftarrow d_i \bigcup \{s_j \in G_{temp}\}$;
    $G_t' \longleftarrow G_t' \bigcup G_{temp}$, $D \longleftarrow D \bigcup \{d_i \mid \forall d_i \in D \cap G_{temp}\}$;
  else
    $Min = Mid$, $\mathcal{P}_t$ =RN-Greedy-MWSC(S_t, D);
  end if
end while
for each node $s_j \in S_t$ **do**
  Find s_j's path in $\mathcal{P}_t$ and calculate s_j's actual destination d_j^t on the path based on the evacuee speeding and t's duration;
  $D_t \longleftarrow D_t \bigcup \{d_j^t\}$, $\mathcal{P}_t^* \longleftarrow \mathcal{P}_t^* \bigcup \{path(s_j, d_j^t)\}$;
end for
return $\mathcal{P}_t^*, D_t$.

Algorithm 1: Route-Network Planning Algorithm.

Construct a route network from S_t to D. *in:* S_t, D *out:* RN
Set $avr = \sum_{d_i \in D} |d_i|/|D|$, $sumpri = \sum_{d_i \in D} priority(d_i)$.
for each $d_i \in D$ **do**
  Use its redundant utilization as its weight, $redutil(d_i) = |d_i| - TH$, where $TH = avr \cdot (sumpri + priority(d_i))/sumpri$ is the threshold of the traveling-person number for an exit.
end for
Set $\mathcal{S} \longleftarrow S_t$, $\mathcal{D} \longleftarrow D$, $RN \longleftarrow \emptyset$.
while $(\mathcal{D} \ne \emptyset)$ **do**
  Select the $d_i \in \mathcal{D}$ with the minimum average redundant utilization ratio, $|redutil(d_i)/|d_i \bigcap \mathcal{S}||$;
  if $(|d_i| > TH)$ **then**
    $d_i \longleftarrow$ {The first TH nearest nodes in d_i}
  end if
  $d_i \longleftarrow d_i \bigcap \mathcal{S}$, $\mathcal{D} \longleftarrow \mathcal{D} - \{d_i\}$, $\mathcal{S} \longleftarrow \mathcal{S} - d_i$, $RN \longleftarrow RN \bigcup \{d_i\}$.
end while
if $(\mathcal{S} \ne \emptyset)$ **then**
  return False;
else
  return RN;
end if

Algorithm 2: Function: RN-Greedy-MWSC Algorithm.

coordinate values. And the edge set E is randomly generated based on the deployment of V to guarantee that the 3D graph is connected: firstly allocate the connecting relationship to each pair of nodes in V; secondly determine whether the graph is connected. If being connected, the edge generation is successful; otherwise, repeat the edge generation process. We consider the following parameters and their variation modes are described as follows:

(i) $n = |V|$: the number of observation nodes in the constrained space;

(ii) $s = |S|$: the number of source nodes;

(iii) $d = |D|$: the number of destination nodes;

(iv) α: an edge's priority is decided by the relevance between the edge and the destinations with different

Table 1: Measurement index of algorithm performance.

Factor	Calculation	Measurement method
Priority	$\max \left\lvert \frac{priority(d_p)}{\lvert d_p \rvert} - \frac{priority(d_q)}{\lvert d_q \rvert} \right\rvert$ $(\forall d_p, d_q \in D)$	A lower priority factor stands for the better performance.
Average length	$\sum (length(d_i, s_j) - SP(d_i, s_j)) / \sum SP(d_i, s_j)$ $(\forall d_i \in D, \forall s_j \in S)$, where $length(d_i, s_j)$ is the path length between d_i and s_j and $SP(d_i, s_j)$ is the length of shortest path between them.	A lower average length factor stands for the better performance.
Global length	$\frac{\lvert \max length(d_i, s_j) - \max SP(d_i, s_j) \rvert}{\max SP(d_i, s_j)}$ $(\forall d_i \in D, \forall s_j \in S)$	A lower global length factor stands for the better performance.

priorities; i.e., the priorities of destinations are randomly valued in the range of $[1, |D|]$;

(v) Influence factor: edge type (β_1) is decided based on the slope of the edge; passibility (β_2) is assumed to be uniform in walking mode, i.e., $\beta_2 = 1$; disaster spreading status ($\beta_3(e_k, t)$) is determined by laws and characteristics of disaster spreading, where we adopt a random value function; travel length ($length(e_k)$) is calculated as the Euclidean distance.

For the evaluation of algorithm performances, we compare our algorithm with the shortest path (SP) algorithm in dynamic and static patterns; i.e., we apply the algorithms in each timeslot under $\mathcal{F}(G_t, t)$ for dynamic pattern; meanwhile, we apply these algorithms in static pattern only under the latest status of disaster spreading $\mathcal{F}(G_T, T)$ and without the consideration of the disaster changes. We discuss about the average behavioral characteristics of these algorithms (denoted as **Dynamic RN-Greedy-MWSC, Dynamic SP, Static RN-Greedy-MWSC, and static SP**).

We will investigate the influence of all the relevant parameters on our algorithms and the parameter setting is based on the scale proportions between the pairs of the space size, escaping evacuee (source) and exits (destination) in practical:

(a) n varies from 200 to 550 by the step of 50 and $s = 30$, $d = 6$.

(b) s varies from 10 to 45 by the step of 5 and $n = 300$, $d = 6$.

(c) d varies from 2 to 16 by the step of 2 and $n = 300$, $s = 30$.

For each parameter setting, we run 100 instances and compute their average for evaluation. Based on the parameters and their variation modes, we study the average characteristic behavior of the proposed algorithm and evaluate their average performance in terms of three items in Table 1.

5.2. Simulation Results and Analysis

5.2.1. Performance on Priority Factor. We first observe the impact of n, s, and d on the *priority factor*. Overall in the three subfigures of Figure 3, the priority factors of the dynamic and static SP with the variation of parameters are all higher than those of the dynamic and static RN-Greedy-MWSC. And dynamic RN-Greedy-MWSC is obviously lower than the static one during the second half of the variation.

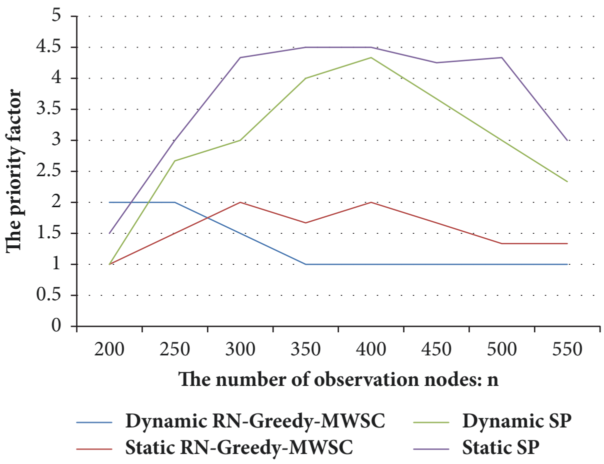
(a) In Figure 3(a), comparing our algorithms with SP algorithms, with the growth of the space scale, the factors in the three compared algorithms are shown in parabolic shapes and the gap between RN-Greedy-MWSCs and SPs becomes bigger. It can be explained as that enlarging the space scale supplies more feasible evacuation paths. Thus, RN-Greedy-MWSCs can perform their advantages on balancing the utilization efficiency among these paths.

Comparing our algorithm in dynamic and static patterns, we can observe that, with the increasing of n, the factor's fluctuation in static RN-Greedy-MWSC is more obvious than that in dynamic RN-Greedy-MWSC and the dynamic one is trending downward on the whole. When $n > 250$, the advantage of dynamic one becomes obvious. It is because when the density of observation vertices deployment becomes larger, the connectivity of the candidate route network is improved, which makes more influence of disaster spreading on each edge. Dynamic RN-Greedy-MWSC considers the influence changes and thus can select paths with better priority-oriented utilization.

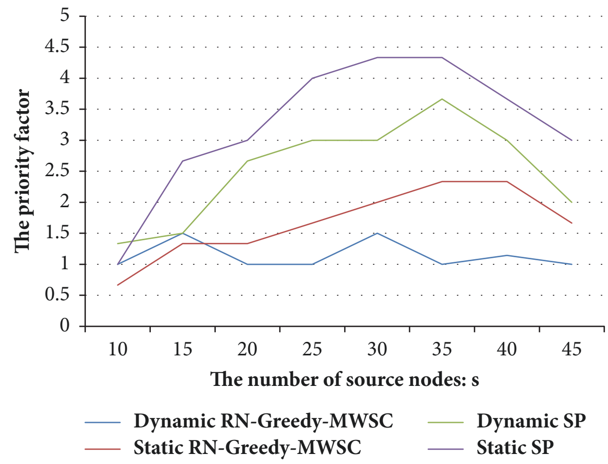
(b) As shown in Figure 3(b), comparing our algorithms with SP algorithms, SP algorithms perform higher than ours on the priority factor and present state of rises and falls. It is because that SP algorithms can guarantee minimizing the route lengths but cannot provide the utilization balance of the routes. Thus, with the changes of source number, the maintaining of the utilization balance may have instability.

Comparing our algorithm in dynamic and static patterns, we also find that when s is larger than 15, the variation trends for the two algorithms become similar and dynamic one maintains the advantage. The number of evacuee s raises the fluctuation in the two algorithms with the peak values at $s = 30$ in dynamic RN-Greedy-MWSC and $s = 35$ in static RN-Greedy-MWSC.

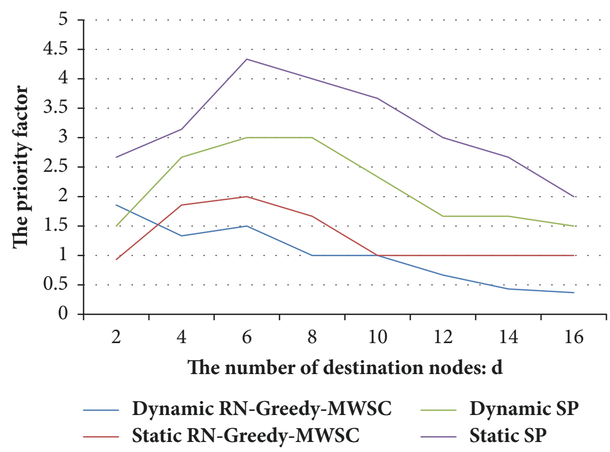
(c) From Figure 3(c), comparing our algorithms with SP algorithms, under a small number of destinations, SP algorithms have undesirable priority factor values but show improvement when $d > 10$. It can be explained that the number of the available and shorter paths can be increased with the growth of destination number, which can raise the efficiency of path utilization. The increasing of the destination



(a) The priority factor versus n



(b) The priority factor versus s



(c) The priority factor versus d

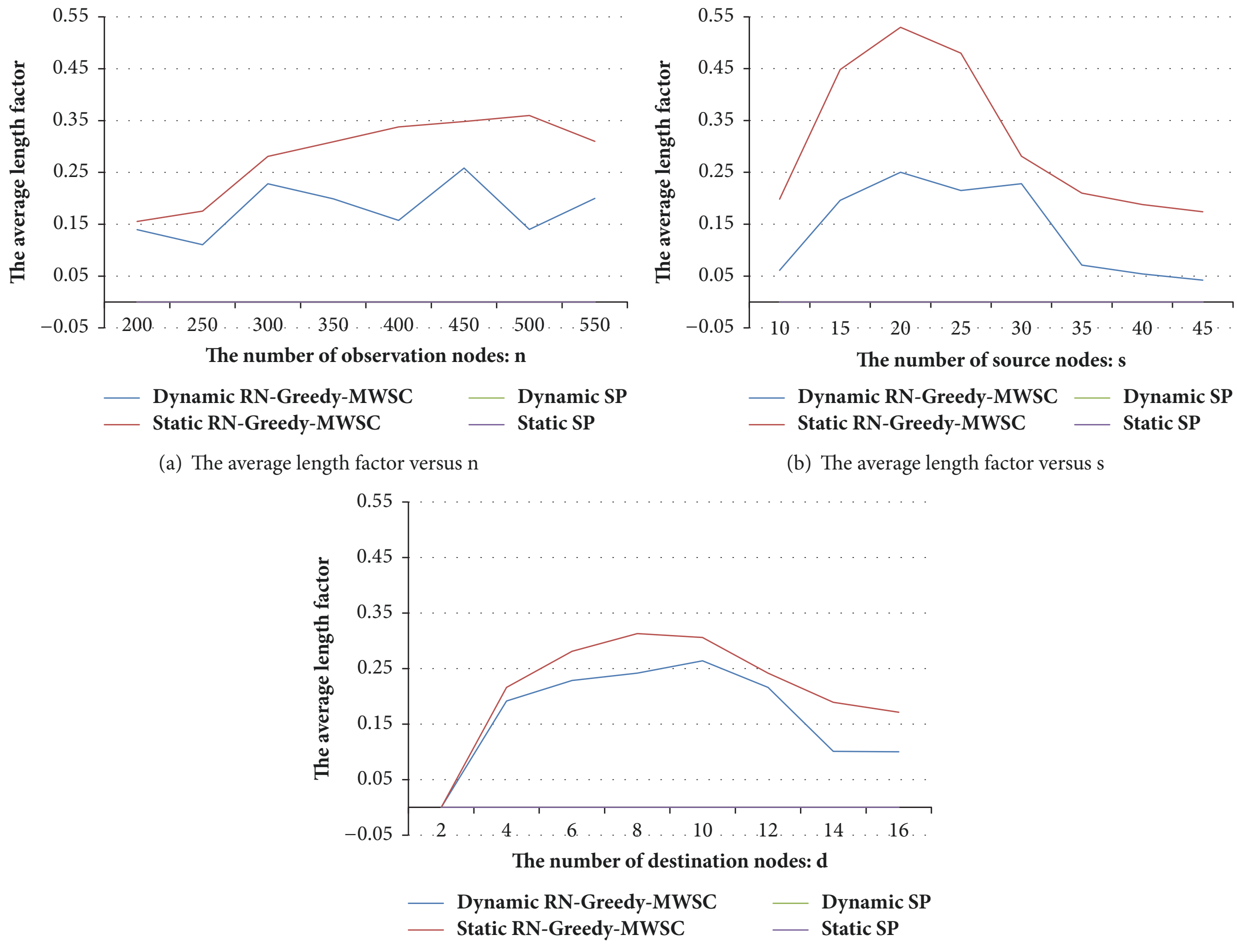
Figure 3: Performance on the priority factor.

number is beneficial to improve the utilization balance of the path planning algorithms.

Comparing our algorithm in dynamic and static patterns, we can learn that the priority factor gained from static RN-Greedy-MWSC can be guaranteed into range $[1, 2]$ and the dynamic one shows a trend of decline and becomes much closer to 0 along with the rise of the exit number. The reason is that with the adjustment of exit utilization based on disaster spreading, the priority-oriented utilization for each exit can be enhanced.

5.2.2. Performance on Average Length Factor. Secondly, we investigate the impact of n, s, and d on the *average length factor*. Note that, for the SP algorithm, the average length factors of dynamic and static SP are both 0 ($= length(d_i, s_j) - SP(d_i, s_j)$) which are the lower bounds of all the path-finding algorithms. These three subfigures show that dynamic and static RN-Greedy-MWSCs have gaps with SP on the average length performance. In Figure 4(a), with the growth of due to the consideration of the balanced utilization of each destination, the path assignment in RN-Greedy-MWSC can be adjusted under the change of the path network topological structure. Based on Figures 4(b) and 4(c), it is worth noting that, with the increasing of the number of sources or destinations, the gap between the average length of the escaping network generated by RN-Greedy-MWSC and that by SP tends to be gradually narrowing.

Comparing dynamic RN-Greedy-MWSC with static RN-Greedy-MWSC, the average length factors of the dynamic one with the variation of parameters are all higher than the static one. Based on Figure 4(a), the average length factor of dynamic RN-Greedy-MWSC is in volatility and that of static RN-Greedy-MWSC is gradually raising along with the rise of n. It is because, with the enlarging of the scale of restricted-space, the global lengths of paths between each pair of source and destination will be increased. From Figures 4(b) and 4(c), we can observe that the number of source nodes s has more influence on this factor of our solutions than the number of destination nodes d. With the increasing of s, the factor of two algorithms presents parabolic shape in the range $[10, 25]$ and the gap between the two algorithms reaches its maximum when $s = 20$. And with the growth of d, the factors of two algorithm both show a parabolic shape in a reasonable range $[0, 0.35]$.



(a) The average length factor versus n

(b) The average length factor versus s

(c) The average length factor versus d

FIGURE 4: Performance on the average length factor.

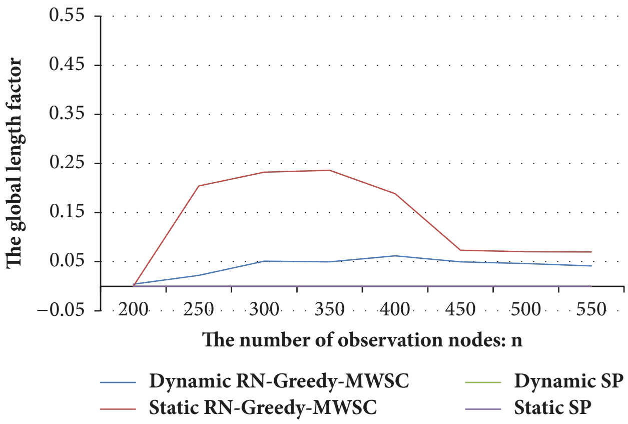
5.2.3. Performance on Global Length Factor. Similar to the average length factor, the global length factors of dynamic and static SPs are both 0 (= $\max length(d_i, s_j) - \max SP(d_i, s_j)$) which are the lower bounds of our algorithms. Seen from Figure 5(a), the difference between the global length obtained from RN-Greedy-MWSCs and that from SPs is getting smaller with the rising of the space scale. And when n is larger than 450, the difference is getting 0 which indicates our strategy can enhance the utilization efficiency of the escaping exits with the minimization guarantee of the global length. On the contrary, as shown in Figures 5(b) and 5(c), the difference becomes larger with the increasing of the source number and could be controlled in a steady state when $s \geq 35$ or $d \geq 10$. This reason is that, with the rise of the source number or destination number, the utilization efficiency of shorter paths can be increased, which can narrow the gap of global path length between SP algorithms and RN-Greedy-MWSCs.

Comparing dynamic RN-Greedy-MWSC with static RN-Greedy-MWSC, seen from Figures 5(a) and 5(c), the difference between the global escaping time consumption obtained from dynamic solution and that from static scheme is getting smaller with the enlarging of space scale or the rising of the evacuee number. And when n is larger than 450 and $d \geq 14$, the difference is approaching 0 which indicates our strategies can enhance the priority utilization efficiency of multiple escaping exits with the complication of the space structure. On the contrary, as shown in Figure 5(b), the difference between the two algorithms is getting larger since the source count is larger than 25, which is also caused by the dynamic influence of the disaster.

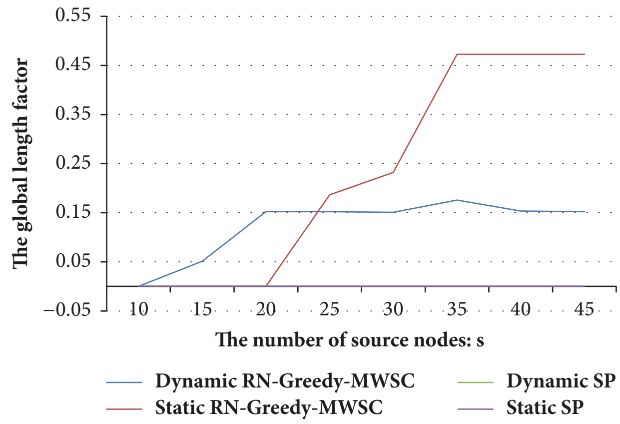
Via the performance evaluations on the three measurement factors, we can draw the conclusions.

(A) The path planning algorithms: firstly, RN-Greedy-MWSC algorithms have more advantages on path utilization than SP algorithms. And the dynamic RN-Greedy-MWSC outperforms the static one in terms of priority factor when the scale of a restricted-space, the scale of sources, and destinations are relatively large; secondly, when evaluating on the average and global path length, the advantage of dynamic RN-Greedy-MWSC maintains signification.

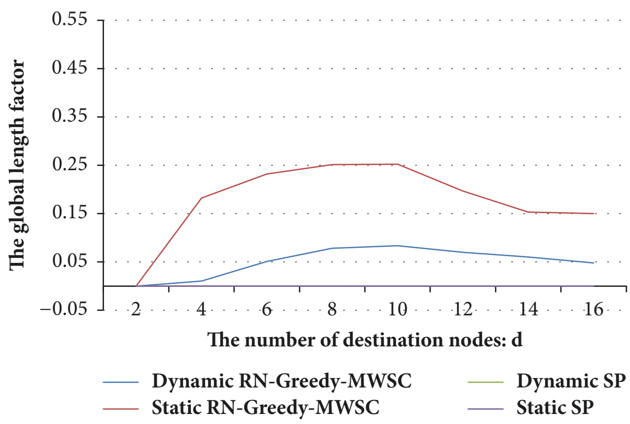
(B) The influence factors: the influence of the scale of sources has more influence on the algorithm performance than that of destinations and space scale.



(a) The global length factor versus n



(b) The global length factor versus s



(c) The global length factor versus d

Figure 5: Performance on the global length factor.

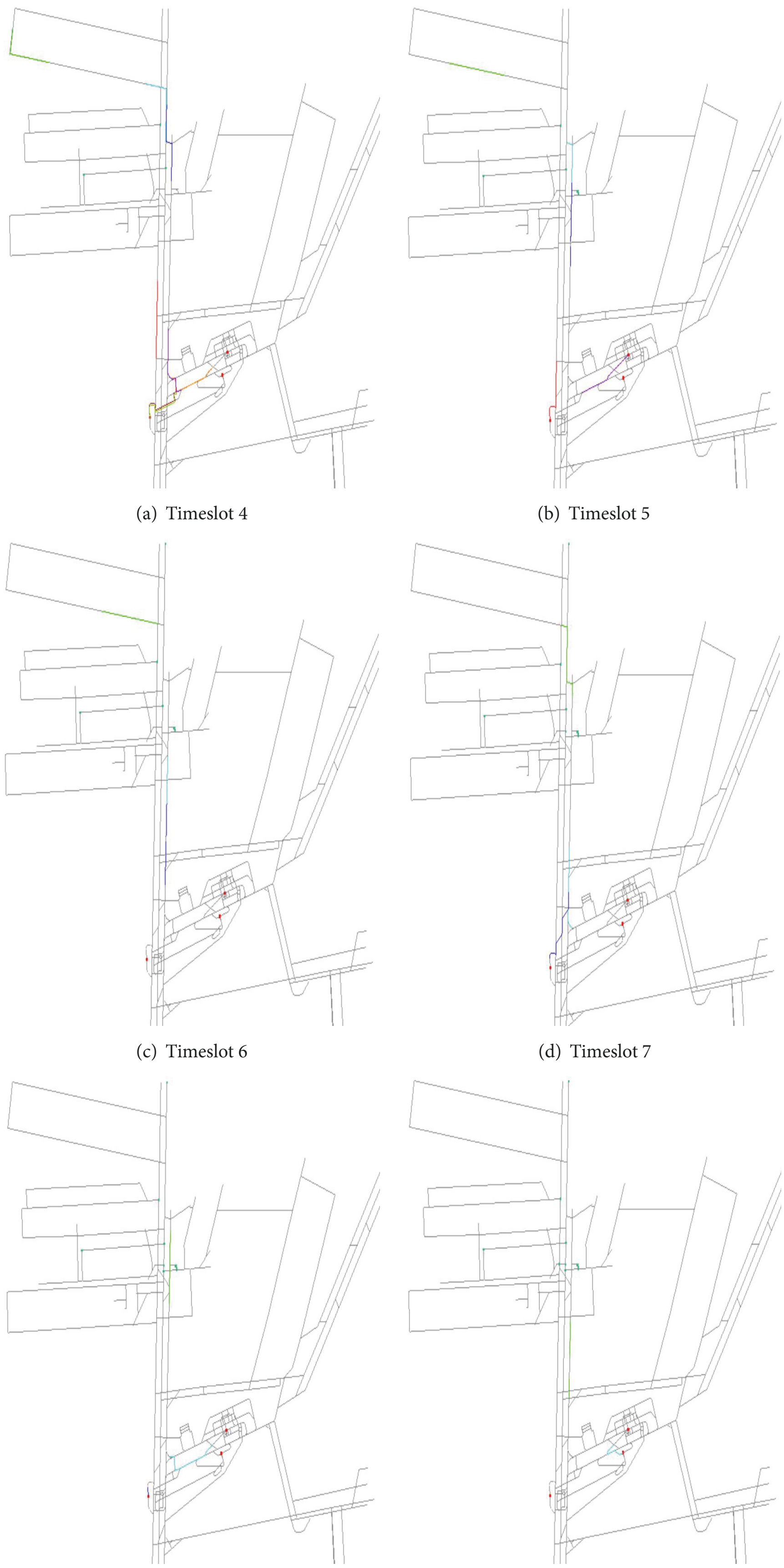
Table 2: Instance experiments results.

	d1	d2	d3	PF	ALF	GLF
Dynamic Algorithm	2	1	7	1.28	0.1321	0.0251
Static Algorithm	3	0	7	∞	0	0

5.3. Application Instance Experiment and Discussion. We further perform an instance experiment on our algorithm in a representative kind of restricted-space application scenarios, underground coal mine water inrush. Here we adopt the 3D model data of a mine area in Zhangjiakou, Hebei Province of China from Kailuan Group Company. We preprocess and transform the original spatial data of the model; i.e., the vertex set in G is composed of all the traverse points, and the edge set of G maintains consistent direction with the original laneway. Water spreading status is obtained by the water spreading algorithm in [29]; i.e., in each timeslot, the prediction data of the water level on all observation points are the input of our algorithms. According to practical requirements, the priorities of three feasible exits, auxiliary shaft (d1), main shaft (d2), and air shaft(d3), are valued as follows: $priority(d1) = 5$, $priority(d2) = 2$, and $priority(d3) = 1.5$.

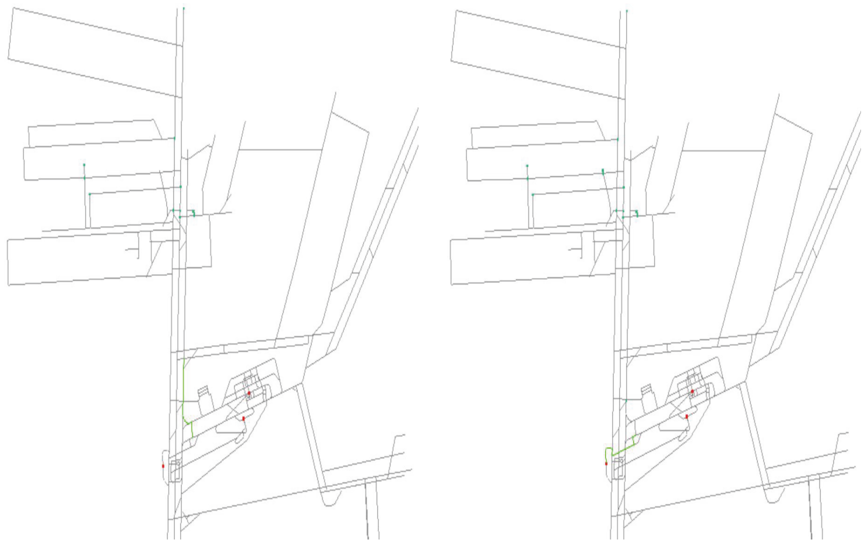
In the visualization results in Figure 6, we present the paths from the 10 evacuation source positions to their temporal destinations from timeslots 4 to 11, and finally all the evacuees successfully escape from the 3 red exits in the last subfigure. Note that the points in blue stand for the submerged part by water inrush, which are all avoided from by the escapers in our algorithms. In Table 2, the evaluation criteria, priority factor, average length factor, and global length factor are represented by PF, ALF, and GLF in short. The number of paths reaching these exits is denoted as d1, d2, and d3, respectively. The parameter results show that the dynamic algorithm outperforms the static one in terms of priority factor in a practical application scene with large scale.

To conclude, our algorithm has advantages on the performance of the escaping exit priority utilization and the evacuation time consumption. And when the scale of space, evacuee, and exits are relatively high, the advantage will be significant. Furthermore, the parameter, the number of evacuees, has more influence on our algorithm than the number of exits on the length factor. Therefore, our strategy can enhance the priority utilization efficiency of the escaping



(a) Timeslot 4
(b) Timeslot 5
(c) Timeslot 6
(d) Timeslot 7
(e) Timeslot 8
(f) Timeslot 9

FIGURE 6: Continued.



(g) Timeslot 10 (h) Timeslot 11

Figure 6: The visualization results of dynamic algorithm.

exits and guarantee a tolerable range of the global escaping time consumption.

6. Conclusions

In this paper, we propose an optimization problem for emergency evacuation planning in restricted-space scenarios, the Multiobjective Dynamic Route Network Planning (MOD-RNP) problem. To minimize the whole evacuation delay and maximize the priority-oriented evacuation efficiency, we design a dynamic and global-optimized strategy to construct the evacuation route network, based on Minimum Weighted Set Cover (MWSC). Based on the performance evaluation, our algorithm outperforms the shortest path algorithm in dynamic pattern on the priority utilization efficiency of the escaping exits, and it can guarantee that the deviation of the global travel length from the shortest one is in a reasonable range. Furthermore, we believe there are lots of room for further investigation and we plan to design distributed dynamic route network planning strategies to contribute or reorganize route network in real time.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This research was supported in part by Beijing Natural Science Foundation (4174090) and Program of Beijing Excellent Talents Training for Young Scholar (2016000020124G056). Professor Wu and Professor Xu were supported in part by China National Scientific and Technical Support Program (2016YFC0801801), China National Natural Science Foundation (41430318, 41272276, 41572222, and 41602262), Beijing Natural Science Foundation (8162036), Fundamental Research Funds for the Central Universities (2010YD02), Innovation Research Team Program of Ministry of Education (IRT1085), and State Key Laboratory of Coal Resources and Safe Mining.

References

[1] Y.-J. Zheng, H.-F. Ling, X.-L. Xu, and S.-Y. Chen, "Emergency scheduling of engineering rescue tasks in disaster relief operations and its application in China," *International Transactions in Operational Research*, vol. 22, no. 3, pp. 503–518, 2015.

[2] L. Özdamar and O. Demir, "A hierarchical clustering and routing procedure for large scale disaster relief logistics planning," *Transportation Research Part E: Logistics and Transportation Review*, vol. 48, no. 3, pp. 591–602, 2012.

[3] M. Lahmar, T. Assavapokee, and S. A. Ardekani, "A dynamic transportation planning support system for hurricane evacuation," in *Proceedings of the Intelligent Transportation Systems Conference, ITSC '06*, pp. 612–617, 2006.

[4] Q. Lu, G. B. George, and S. Shekhar, "Capacity constrained routing algorithms for evacuation planning: a summary of results," in *Proceedings of the International Symposium on Advances in Spatial & Temporal Databases*, vol. 3633, 8, pp. 291–307, 2005.

[5] S. Kim, B. George, and S. Shekhar, "Evacuation route planning: Scalable heuristics," in *Proceedings of the 15th ACM International Symposium on Advances in Geographic Information Systems, GIS '07*, vol. 20, 8, pp. 1–8, November 2007.

[6] A. H. Sampaio and S. Urrutia, "New formulation and branch-and-cut algorithm for the pickup and delivery traveling salesman problem with multiple stacks," *International Transactions in Operational Research*, vol. 24, no. 1-2, pp. 77–98, 2017.

[7] F. Hernandez, M. Gendreau, and J.-Y. Potvin, "Heuristics for tactical time slot management: a periodic vehicle routing problem view," *International Transactions in Operational Research*, vol. 24, no. 6, pp. 1233–1252, 2017.

[8] M. Kammoun, H. Derbel, M. Ratli, and B. Jarboui, "An integration of mixed VND and VNS: the case of the multivehicle covering tour problem," *International Transactions in Operational Research*, vol. 24, no. 3, pp. 663–679, 2017.

[9] P. Chen, B. Golden, X. Wang, and E. Wasil, "A novel approach to solve the split delivery vehicle routing problem," *International Transactions in Operational Research*, vol. 24, no. 1-2, pp. 27–41, 2017.

[10] Y.-J. Zheng, S.-Y. Chen, and H.-F. Ling, "Evolutionary optimization for disaster relief operations: a survey," *Applied Soft Computing*, vol. 27, pp. 553–566, 2015.

[11] S. Kim and S. Shekhar, "Contraflow network reconfiguration for evacuation planning: a summary of results," in *Proceedings of the 13rd ACM International Symposium on Geographic Information Systems, ACM-GIS '05*, pp. 250–259, 2005.

[12] G. J. Lim, S. Zangeneh, M. Reza Baharnemati, and T. Assavapokee, "A capacitated network flow optimization approach for short notice evacuation planning," *European Journal of Operational Research*, vol. 223, no. 1, pp. 234–245, 2012.

[13] A. Takizawa, M. Inoue, and N. Katoh, "An emergency evacuation planning model using the universally quickest flow," *The Review of Socionetwork Strategies*, vol. 6, no. 1, pp. 15–28, 2012.

[14] N. Baumann and M. Skutella, "Solving evacuation problems efficiently Earliest arrival flows with multiple sources," in *Proceedings of the 47th Annual IEEE Symposium on Foundations of Computer Science, FOCS '6*, pp. 399–408, 2006.

[15] C. E. Dunn and D. Newton, "Optimal routes in GIS and emergency planning applications," *Area*, vol. 24, no. 3, pp. 259–267, 1992.

[16] T. Yamada, "A network flow approach to a city emergency evacuation planning," *International Journal of Systems Science*, vol. 27, no. 10, pp. 931–936, 1996.

[17] T. J. Cova and J. P. Johnson, "A network flow model for lane-based evacuation routing," *Transportation Research Part A: Policy and Practice*, vol. 37, no. 7, pp. 579–604, 2003.

[18] N. Kamiyama, N. Katoh, and A. Takizawa, "An efficient algorithm for evacuation problems in dynamic network flows with uniform arc capacity," *Lecture Notes in Computer Science*, vol. 4041, pp. 231–242, 2006.

[19] N. Kamiyama, N. Katoh, and A. Takizawa, "An efficient algorithm for the evacuation problem in a certain class of networks with uniform path-lengths," *Discrete Applied Mathematics: The Journal of Combinatorial Algorithms, Informatics and Computational Sciences*, vol. 157, no. 17, pp. 3665–3677, 2009.

[20] J. W. Wang, H. F. Wang, W. J. Zhang, W. H. Ip, and K. Furuta, "Evacuation planning based on the contraflow technique with consideration of evacuation priorities and traffic setup time," *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 1, pp. 480–485, 2013.

[21] O. L. Huibregtse, S. P. Hoogendoorn, A. Hegyi, and M. C. Bliemer, "A method to optimize evacuation instructions," *OR Spectrum*, vol. 33, no. 3, pp. 595–627, 2011.

[22] M. Goodwin, O.-C. Granmo, and J. Radianti, "Escape planning in realistic fire scenarios with Ant Colony Optimisation," *Applied Intelligence*, vol. 42, no. 1, pp. 24–35, 2014.

[23] M. Goerigk, K. Deghdak, and P. Heßler, "A comprehensive evacuation planning model and genetic solution algorithm," *Transportation Research Part E: Logistics and Transportation Review*, vol. 71, pp. 82–97, 2014.

[24] M. Manley, Y. S. Kim, K. Christensen, and A. Chen, "Airport Emergency Evacuation Planning: An Agent-Based Simulation Study of Dirty Bomb Scenarios," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 46, no. 10, pp. 1390–1403, 2016.

[25] S.-Q. Tong, N. Wang, and N.-Q. Song, "Emergency evacuation capability evaluation and optimization for an offshore airport: The case of Dalian Offshore Airport, Dalian, China," *Safety Science*, vol. 92, pp. 128–137, 2017.

[26] H. Liu, B. Xu, D. Lu, and G. Zhang, "A path planning approach for crowd evacuation in buildings based on improved artificial bee colony algorithm," *Applied Soft Computing*, vol. 68, pp. 360–376, 2018.

[27] N. Al Theeb and C. Murray, "Vehicle routing and resource distribution in postdisaster humanitarian relief operations," *International Transactions in Operational Research*, vol. 24, no. 6, pp. 1253–1284, 2017.

[28] M. R. Garey and D. S. Johnson, "'Strong' NP-completeness results: motivation, examples, and implications," *Journal of the ACM*, vol. 25, no. 3, pp. 499–508, 1978.

[29] C.-P. Li, Z.-X. Li, Y.-X. Zheng, Li. Xin, D.-Y. Hou, and Y.-D. Zhou, "Three-dimensional dynamic simulation modeling of water inrushes in underground mines," *Journal of University of Science and Technology Beijing*, vol. 35, no. 2, pp. 140–147, 2013.