

Research Article

Dynamic Path Optimization with Real-Time Information for Emergency Evacuation

Huajun Zhang , Qin Zhao , Zihui Cheng, Linfan Liu, and Yixin Su

School of Automation, Wuhan University of Technology, Wuhan, China

Correspondence should be addressed to Qin Zhao; zhaoqin@whut.edu.cn

Received 12 April 2021; Revised 14 July 2021; Accepted 28 July 2021; Published 3 August 2021

Academic Editor: Mohammad Yazdi

Copyright © 2021 Huajun Zhang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In order to find the optimal path for emergency evacuation, this paper proposes a dynamic path optimization algorithm based on real-time information to search the optimal path and it takes fire accident as an example to introduce the algorithm principle. Before the accidents, it uses the Dijkstra algorithm to get the prior evacuation network which includes evacuation paths from each node to the exit port. When the accidents occur, the evacuees are unable to pass through the passage where the accident point and the blocking point are located, then the proposed method uses the breadth-first search strategy to solve the path optimization problem based on the prior evacuation network, and it dynamically updates the evacuation path according to the real-time information. Because the prior evacuation network includes global optimal evacuation paths from each node to the exit port, the breadth-first search algorithm only searches local optimal paths to avoid the blockage node or dangerous area. Because the online optimization solves a local pathfinding problem and the entire topology optimization is an offline calculation, the proposed method can find the optimal path in a short time when the accident situation changes. The simulation tests the performances of the proposed algorithm with different situations based on the topology of a building, and the results show that the proposed algorithm is effective to get the optimal path in a short time when it faces changes caused by the factors such as evacuee size, people distribution, blockage location, and accident points.

1. Introduction

The most important thing for emergency evacuation is to guide people to move from dangerous areas to safe areas in the shortest time. Facing emergencies such as hurricanes, tsunamis, fires, and poisonous gas leakage [1], some methods were proposed to find the optimal path according to the prior information [2, 3], and the methods are not suitable for the emergency evacuation when the situations are significantly different from the prior information. Studies have shown that fire is the most frequent accident among the above emergencies [4–6], and it gets more attention from researchers since fire accidents frequently cause huge losses to mankind; then, this paper takes fire accident as an example to introduce the proposed algorithm principle. The evacuation environment includes route topology, exit location, evacuee size, and people distribution, and changes of each element bring different evacuation paths. Fire accidents

often happen suddenly and unpredictably, and the static path planning algorithms have limited application in practice since the prior environment information is different from the real-time environment information during emergencies. According to the studies of some fire event cases [7–9], chaotic and untimely evacuation is the important reason for the high casualty rate. In crowded buildings such as shopping malls, supermarkets, and large cruise, people always move without any regular pattern, and it is necessary to search evacuation paths according to exit locations, real-time evacuee locations, and evacuee size [10, 11]. It is dangerous for evacuees to pass a passageway that is full of smoke, and the smoke spreading changes the route topology during the evacuation. The old path is not fit for the new situation when some passageways change, and it is a time-consuming process to find a new optimal path. In order to find the evacuation path in a short time, some methods [12] simplified the route topology to reduce the

computation time, and the incomplete dynamic route topology made the new path different from the shortest one. With the development of stochastic optimization, some intelligent algorithms such as ant colony algorithm [13–15], particle swarm algorithm [16–18], bee colony algorithm [19–21], and genetic algorithm [22] or greedy algorithm [23] were used to find the dynamic path. When route topography and constraints changed, intelligent algorithms took the old path as a feasible solution for dynamic path searching. Although intelligent algorithms are able to search for optimal solutions globally, they converge slowly when facing dynamic evacuation problems.

Recently, the dynamic evacuation problem has become a hot research topic, and many scholars try to get the shortest path according to dynamic situations. The authors in references [24–27] built the intelligent integrated fire rescue system which updated optimal paths by using the wireless sensor or vision camera to detect people locations. However, the wireless sensors and vision cameras are difficult to get accurate dynamic information about the people, and the online computation is also complex and time consuming. In order to get the shortest paths in a short time, basic and improved Dijkstra algorithms are the most popular methods used to search the optimal paths [28–30]. Beside the path length, safety is also important for people during the evacuation, and the authors references [31–33] took the safest and shortest paths as the cost function to search the optimal paths. Facing the dynamic situation, the authors in references [34–37] used dynamic planning to search for the optimal paths and informed everyone on the ship how to move toward the exit. Neural networks are good at learning the hidden knowledge from sample data, and the authors in [38–41] used neural networks to decide which paths are available for people to evacuate dynamically. Because the smog keeps spreading during the fire accident, feasible evacuation paths should consider the future changes of dangerous areas. The authors in [42–44] tried to predict the dangerous area and searched the optimal paths step by step, and the prediction performances influenced the effectiveness of the evacuation significantly.

Effective emergency evacuation paths are important for the safety of personal in accidents, and the basic requirement is that the commander gets the optimal paths to fit the dynamic changes in a short time. This paper proposes a dynamic path optimization method which includes offline search and online search. The offline optimization gets the prior evacuation network which includes evacuation paths from each node to the exit port, and the online optimization searches the local optimal paths based on the prior evacuation network when the accidents happen. The changes from accident locations, dangerous areas, and blocking paths call the online optimization algorithm to search the dynamic optimal paths. Online optimization quickly plans the evacuation paths to adapt the real-time situations.

This paper is organized as follows: Section 2 introduces the method of establishing the network topology of the building. Section 3 details the proposed dynamic optimization algorithm that uses offline optimization and online optimization to search the optimal evacuation paths. The

simulation results and analysis are in Section 4. Section 5 summaries the conclusions.

2. Evacuation Environment Model

The emergency evacuation model mainly includes microscopic model and macroscopic model [45]. The microscopic model focuses on the study of individual behavior characteristics, and the path optimization algorithm simulates human individual behavior and mental activities to select evacuation paths. Typical microscopic models include the social force model, cellular automata model, and lattice gas model [46]. Because the microscopic model uses less overall information about the environment, it always gets local feasible paths and ignores the global optimal paths. The local feasible paths are able to guide people to move toward the exit, but it is different from the optimal solution that considers the evacuation time and personal safety during the evacuation process. In addition, the optimization algorithms based on the microscopic model converge slowly, and the online pathfinding requires fast convergence algorithms [47]. The macroscopic model pays attention to the people flow of each evacuation path at different moments based on global information. People move at different speeds when the density of people changes, and the move speeds are macroviews of individual behavior. The relationship between move speed and people density is summarized by the characteristics of individual movement and saved as a piecewise function. Since the optimization outputs based on the macroscopic model are global optimal solutions, the paths are more effective for emergency evacuation.

In the macroscopic model, paths and intersections of the building are regarded as the edges and nodes of a graph. The macroscopic model is a weighted graph as follows:

$$G = (V, E, W), \quad (1)$$

where V is the set of nodes, $V = \{v_i \mid i \in [1, n]\}$, in which n is the total number of the network nodes. E is the set of graph edges, $E = \{e_{ij} \mid i, j \in V\}$. W is the set of weights of each edge. Usually, corridors or stairways in a building are set as edges, and the endpoints of corridors are set as nodes, and the weight of each edge is the actual distance between the connected nodes. In the large-scale macroscopic model, optimization also requires complex computation due to a large number of nodes and edges.

This paper uses macroscopic modeling principles to establish the model of an underground shopping mall and related passage spaces. This shopping mall is the middle floor of a three-floor structure, the bottom floor is the subway station, and the top floor has several urban traffic roads. The macroscopic model represents the fire evacuation environment of the surrounding area, and its graph is shown in Figure 1. The model is a multioutlet network, where V is a set of ordinary nodes and $V = \{N_1, N_2, \dots, N_{22}\}$ and O is a set of safe exit nodes and $O = \{O_1, O_2, O_3, O_4\}$. Each weight is the actual distance between adjacent nodes, and the weight values are shown in Table 1.

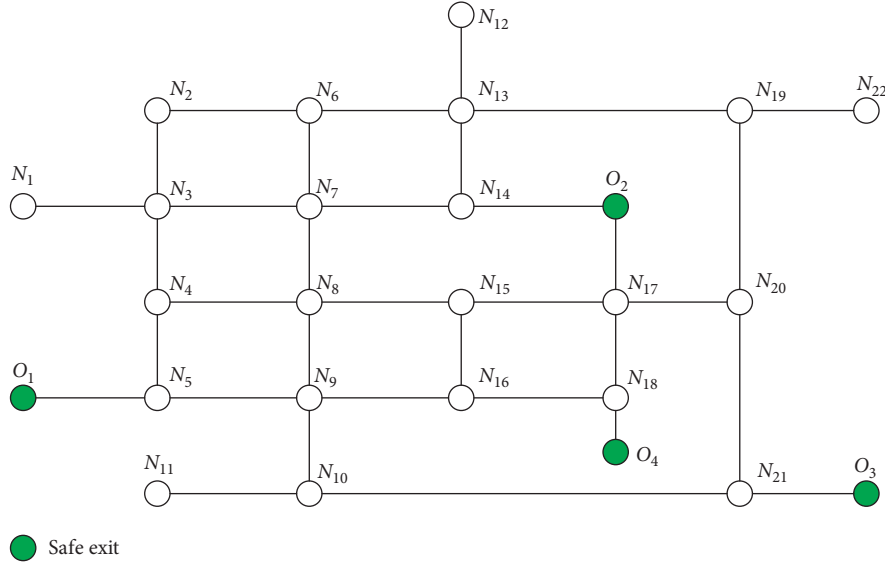


FIGURE 1: Network model of the underground shopping mall.

TABLE 1: The weight values between adjacent nodes.

Node number	Weight value (m)
1	$w_{1,3} = 32$
2	$w_{2,3} = 7, w_{2,6} = 11$
3	$w_{3,1} = 32, w_{3,2} = 7, w_{3,4} = 21, w_{3,7} = 5$
4	$w_{4,3} = 21, w_{4,5} = 13, w_{4,8} = 8$
5	$w_{5,4} = 13, w_{5,9} = 14, w_{5,O1} = 6$
6	$w_{6,2} = 11, w_{6,7} = 43, w_{6,13} = 64$
7	$w_{7,3} = 5, w_{7,6} = 43, w_{7,8} = 17, w_{7,14} = 11$
8	$w_{8,4} = 8, w_{8,7} = 17, w_{8,9} = 4, w_{8,15} = 11$
9	$w_{9,5} = 14, w_{9,8} = 4, w_{9,10} = 44, w_{9,16} = 5$
10	$w_{10,9} = 44, w_{10,11} = 8, w_{10,21} = 92$
11	$w_{11,10} = 8$
12	$w_{12,13} = 8$
13	$w_{13,6} = 64, w_{13,12} = 8, w_{13,14} = 51, w_{13,19} = 13$
14	$w_{14,7} = 11, w_{14,13} = 51, w_{14,O2} = 3$
15	$w_{15,8} = 11, w_{15,16} = 13, w_{15,17} = 41$
16	$w_{16,9} = 5, w_{16,15} = 13, w_{16,18} = 37$
17	$w_{17,15} = 41, w_{17,18} = 26, w_{17,20} = 81, w_{17,O2} = 14$
18	$w_{18,16} = 37, w_{18,17} = 26, w_{18,O4} = 99$
19	$w_{19,13} = 13, w_{19,20} = 52, w_{19,22} = 14$
20	$w_{20,17} = 81, w_{20,19} = 52, w_{20,21} = 34$
21	$w_{21,10} = 92, w_{21,20} = 34, w_{21,O3} = 12$
22	$w_{22,19} = 14$

3. Dynamic Path Optimization

The proposed dynamic path optimization includes offline optimization and online optimization. With prior information of the evacuation model shown in Figure 1, the offline optimization is used to search for the optimal paths that contain evacuation routes from each node to exit nodes. The online optimization dynamically searches the optimal paths when real-time information is different from the prior information. The online optimization searches the local areas and converges fast, and it gets new optimal paths that fit the real-time requirement in a short time.

3.1. Offline Prior Evacuation Path Network. Before the fire accident, it is necessary to get the shortest evacuation path from each node to each exit for the building. This search is an offline process that uses the Dijkstra algorithm to get the prior evacuation path network that stores the node number of each evacuation path. The Dijkstra algorithm is a typical algorithm for searching the shortest path between nodes in a graph, and it has been widely used to search for the shortest evacuation paths in fire incidents. The flow of the standard Dijkstra algorithm is shown as Algorithm 1. The operation $u \leftarrow \text{vertex in } Q \text{ with min dist } [u]$ means the algorithm searches for the vertex u in the vertex set Q that has the least distance. Operation $\text{length}(u, v)$ returns the length of the edge connecting two adjacent nodes u and v .

The variable expand is the length of the path from the source node to the node v via u . The principle used to determine whether the node v satisfies the slack condition is as follows:

$$d_{(u,v)} > d_{(u,q)} + d_{(q,v)}, \quad (2)$$

where $d_{(u,v)}$ is the distance between node u and node v . Equation (2) shows that the distance from node u to node v via node q is shorter than the old distance between node u and node v , and then the path $p(u, v)$ is updated as follows:

$$p(u, v) = \{u, q, v\}. \quad (3)$$

According to the Dijkstra algorithm, we get the evacuation path network of the building shown in Figure 1. The network includes the shortest path from each node to each exit. Since the network is too large to display, Table 2 describes the nodes from node 3 to the four exits and the paths are shown in different colors in Figure 2. If people are at node N_3 , they will use the information shown in Table 2 to quickly escape to the safety exit, so this network is the prior information for the people to escape. When the fire environment is the same as the offline optimized hypothetical

```

create vertex set Q
for each vertex v in Graph:
  dist [v] ← Infinity
  path [v] ← Undefined
  add v to Q
dist [source] ← 0
while Q is not empty:
  u ← vertex in Q with min dist [u]
  remove u from Q
  for each neighbor v of u:
    expand ← dist [u] + length (u, v)
    if expand < dist [v]:
      dist [v] ← expand
      path [v] ← u
return dist [], path []

```

ALGORITHM 1: Offline evacuation path optimization.

TABLE 2: Evacuation paths from node 3 to the exits.

Start node and exit node	Shortest path	Distance (m)
N_3-O_1	$\{N_3, N_4, N_5, O_1\}$	40
N_3-O_2	$\{N_3, N_7, N_{14}, O_2\}$	19
N_3-O_3	$\{N_3, N_7, N_{14}, O_2, N_{17}, N_{20}, N_{21}, O_3\}$	160
N_3-O_4	$\{N_3, N_7, N_{14}, O_2, N_{17}, N_{18}, O_4\}$	158

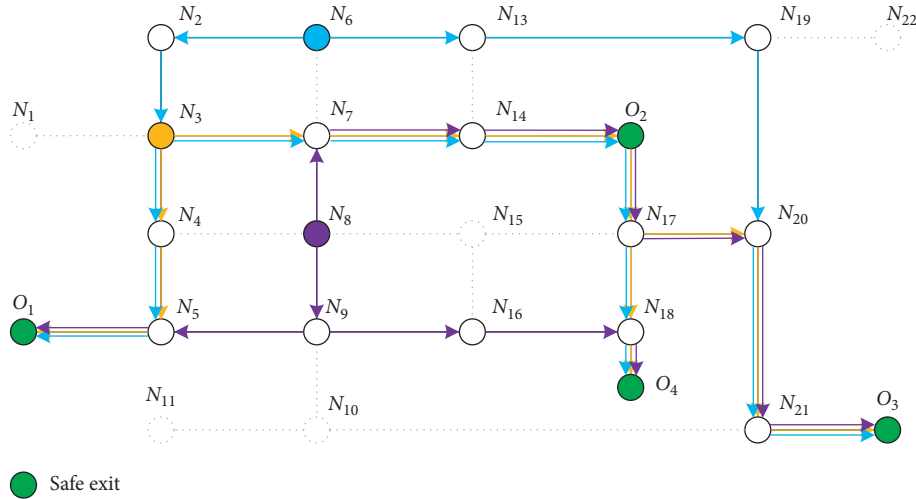


FIGURE 2: The paths from node 3 and node 8 to different exits in the prior evacuation network.

environment, people can move according to the prior evacuation network.

3.2. Online Local Path Optimization. The location and time of the fire are unpredictable; the distribution of people inside the building is also changing. The prior evacuation network needs to be modified when some edges are blocked or in a dangerous area. This modification only adjusts the nodes of the paths that are affected by the fire, and it is a local path optimization based on the prior evacuation network. Because the fire is changing during the evacuation, then the online local path optimization keeps searching for the new solutions. In the network graph model $G = (V, E, W)$, the

evacuation path $p(i, j)$ from the start node i to the end node j is as follows:

$$p(i, j) = \{N_i, N_r, \dots, N_s, N_j\}, \quad (4)$$

where N_i , N_r , N_s , and N_j are the node number. The total distance $\text{dist}\{p(i, j)\}$ of the evacuation path $p(i, j)$ is

$$\text{dist}\{p(i, j)\} = \sum_{t=i}^j w_{(t,t+1)}, \quad (5)$$

where $w_{(t,t+1)}$ is the actual distance between node t and node $t + 1$. The cost function of the dynamic optimization is to search for the paths that do not include the impassable

nodes. Suppose the real-time impassable nodes set is IV , then the online optimization problem is as follows:

$$\min \sum_{t=i}^j w_{(t,t+1)}, \quad t \in V, t \notin IV. \quad (6)$$

Equation (6) is a constraint optimization problem; the optimization result $p(i, j)$ only includes the safe and passable nodes. Suppose PNet is the prior evacuation path network, NBNnode is the neighbor nodes of the impassable nodes in IV . The online optimization starts to search for the optimal paths when the PNet is not fit for the real-time situation, and the online optimization algorithm flow is shown as Algorithm 2.

This paper uses the online dynamic optimization algorithm to search for the real-time optimal paths for the graph in Figure 1. Suppose N_7 is in the fire area and the impassable node set $IV = \{N_7\}$. According to the dynamic optimization algorithm in Algorithm 2, the SNode stores the starting nodes of the prior paths that contains N_7 , and the nodes of SNode are as follows:

$$\begin{aligned} \text{SNode}(1) &= \{N_{12}, N_{13}, N_{14}, N_{17}, N_{19}, N_{20}, N_{22}\}, \\ \text{SNode}(2) &= \{N_1, N_2, N_3, N_4, N_5, N_6, N_8, N_9, N_{10}, N_{11}, N_{15}, N_{16}\}, \\ \text{SNode}(3) &= \{N_1, N_2, N_3, N_4, N_5, N_8, N_9, N_{10}, N_{11}\}, \\ \text{SNode}(4) &= \{N_1, N_2, N_3, N_6\}, \end{aligned} \quad (7)$$

where $\text{SNode}(i)$ contains the starting nodes of the prior paths that are from each node to O_i via N_7 . Because the paths from the nodes in SNode to exits are impassable, then these paths are adjusted dynamically. N_3, N_6, N_8 , and N_{14} are neighbor nodes of N_7 ; then, the nodes of NBNnode are as follows:

$$\begin{aligned} \text{NBNnode}(1) &= \{N_{14}\}, \\ \text{NBNnode}(2) &= \{N_3, N_6, N_8\}, \\ \text{NBNnode}(3) &= \{N_3, N_8\}, \\ \text{NBNnode}(4) &= \{N_6\}, \end{aligned} \quad (8)$$

where $\text{NBNnode}(i)$ contains the neighbor nodes of N_7 in $\text{SNode}\{i\}$. Because the nodes of NBNnode are around node N_7 , the paths from these nodes to the exits help the people to avoid the impassable node N_7 . We use the breadth-first search (BFS) algorithm to search for the optimal paths from the nodes of SNode to the exits. Since the paths from each node to the exits are stored in the prior evacuation network, the BFS gets the shortest path from a node when it visits the node. This prior evacuation path network reduces the search time significantly, and it only needs a little lookup time to get the path information when it visits a node. The BFS visits nodes of NBNnode and checks whether the shortest paths contain impassable node. If path from node N_k to exit O_h does not contain the impassable node, then the new path from node N_i to exit O_h is updated as follows:

$$p(N_i, O_h) = p(N_i, N_k) + p(N_k, O_h), \quad (9)$$

where node $N_i \in \text{SNode}$, $N_k \in \text{NBNnode}$, and O_h is the exit node. The paths from nodes of SNode to the exit nodes are

updated according to equation (9) step by step. Although the new paths avoid impassable node, some paths are not the shortest, and it is necessary to fine-tune the suboptimal paths. Suppose DPNode is the set of the new dynamic paths, and one of the new paths is

$$p(N_i, O_h) = \{N_i, N_r, \dots, N_k, N_t, \dots, N_n, O_h\}, \quad (10)$$

where $N_i, N_r, \dots, N_n \notin IV$. Nodes around the node of $p(N_i, O_h)$ are used to check whether there is a much shorter path; if a node N_α around N_k satisfies the inequality as follows:

$$w(N_k, N_\alpha) + \text{dist}(N_\alpha, O_h) < w(N_k, N_t) + \text{dist}(N_t, O_h), \quad (11)$$

then the new shorter path is as follows:

$$p'(N_i, O_h) = p(N_i, N_\alpha) + p(N_\alpha, O_h). \quad (12)$$

The real-time information triggers the online dynamic optimization to search for the new shortest paths according to the above algorithm principle. In Figure 1, suppose node N_7 is in the fire area, there are 35 paths that contain N_7 . Different paths are shown in Figure 3. Path₁ is the prior shortest evacuation path from node N_6 to exit O_2 , Path₂ is a feasible path that avoids node N_7 , and Path₃ is the final shortest path obtained by global search. It takes 11 ms to update the shortest paths from nodes in SNode to the exits.

4. Simulation Results and Analysis

The more nodes in the network model, the more time it takes to search for evacuation paths in real-time. This paper establishes a network model with 2000 nodes to compare the performances of the proposed dynamic optimization and the traditional Dijkstra algorithm. Suppose the set of safe exit nodes is $O = \{N_7, N_{77}, N_{777}\}$ and each node has two edges connecting to other nodes, the weight of each edge is randomly assigned a positive value between 0 and 100.

Let node N_{18} be in the fire area; both the proposed dynamic algorithm and the Dijkstra algorithm are used to search for the shortest paths from node N_{1225} to exit N_7 and N_{777} . The path optimization results are shown in Table 3. The M1 is the method that uses the Dijkstra algorithm to search for the shortest path offline. M2 and M3 are the methods that use the proposed dynamic algorithm and the Dijkstra algorithm to search for the shortest path according to the real-time information as node N_{18} is in the fire area. M2 and M3 are online searching that avoid passing through node N_{18} . The results show that the evacuation paths of the online searching algorithms are longer than those of the offline algorithm; this difference is due to the fact that the online search algorithms need to avoid node N_{18} . M2 gets the same path as M3 in online path searching, and it shows that the proposed dynamic algorithm has the ability to get the shortest path.

The node from N_1 to N_{200} is set as a fire point step by step, and the curve of the number of nodes in SNode is shown in Figure 4. Different nodes cause different sizes of SNode, and the online searching time is also shown in

```

for  $p$  in PNet
  for  $d$  in IV
    if path  $p$  contains node  $d$ 
      SNode  $\leftarrow$  the first node of path  $p$ 
    for  $N_i$  in NBNode
       $p(N_i, O_h)$  is updated by the breadth-first search algorithm
    for  $N_k$  in SNode
      update the path of DPNode as  $p(N_k, O_h) = p(N_k, N_i) + p(N_i, O_h)$ 
    for  $p$  in DPNode
      for  $N_r$  in path  $p$ 
         $p \leftarrow$  check the nodes around  $N_r$  whether  $p$  is the shortest
    return DPNode

```

ALGORITHM 2: Online dynamic path optimization.

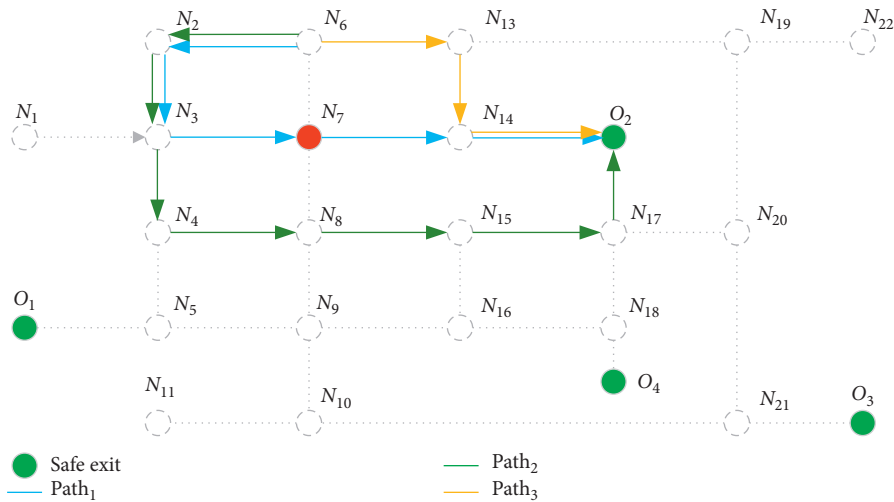
FIGURE 3: The paths from node N_6 to exit O_2 updated by the proposed dynamic algorithm.

TABLE 3: Evacuation paths of different algorithms.

Start-end	Method	Evacuation path	Distance
$N_{1225}-N_7$	M1	$\{N_{1225}, N_{564}, N_{18}, N_{1587}, N_{113}, N_{1210}, N_{653}, N_{1030}, N_{334}, N_{1503}, N_{51}, N_{1289}, N_{1201}, N_7\}$	291
	M2	$\{N_{1225}, N_{1631}, N_{801}, N_{979}, N_{1668}, N_{1580}, N_{432}, N_{1218}, N_7\}$	324
	M3	$\{N_{1225}, N_{1631}, N_{801}, N_{979}, N_{1668}, N_{1580}, N_{432}, N_{1218}, N_7\}$	324
$N_{1225}-N_{777}$	M1	$\{N_{1225}, N_{564}, N_{18}, N_{474}, N_{761}, N_{558}, N_{777}\}$	234
	M2	$\{N_{1225}, N_{564}, N_{1372}, N_{607}, N_{895}, N_{1938}, N_{197}, N_{777}\}$	279
	M3	$\{N_{1225}, N_{564}, N_{1372}, N_{607}, N_{895}, N_{1938}, N_{197}, N_{777}\}$	279

Figure 4. The curves in Figure 4 show that the online searching time is positively correlated with the size of SNode.

With the same nodes in SNode, this paper compares the online searching time of the proposed dynamic algorithm with that of the Dijkstra algorithm, and the results are shown in Figure 5. The comparison result shows that the proposed dynamic algorithm searches much faster than the Dijkstra algorithm all the time. The Dijkstra algorithm executes a global optimization when facing a new order to search for a path from the node in SNode to the exit; it takes a long time

to find out the shortest path. The proposed dynamic algorithm takes advantage of the prior evacuation path and only visits the nodes around the fire area to get the local shortest paths. Because the proposed dynamic algorithm does not search each node of the model, it is more effective than the Dijkstra algorithm. Let t_p denote the online searching time of the proposed dynamic algorithm and t_d denote the online searching time of the Dijkstra algorithm; the ratio of t_p to t_d is also shown in Figure 5. The maximum ratio is 0.92, and the minimum ratio is 0.02, and there are some ratios that have a

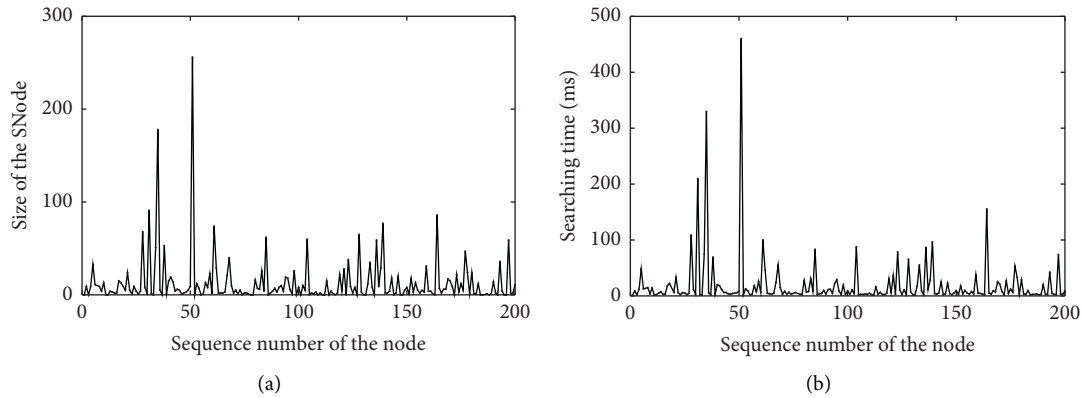


FIGURE 4: The performance of the proposed dynamic algorithm: (a) relationship between the size of SNode and node number; (b) relationship between the searching time and the node number.

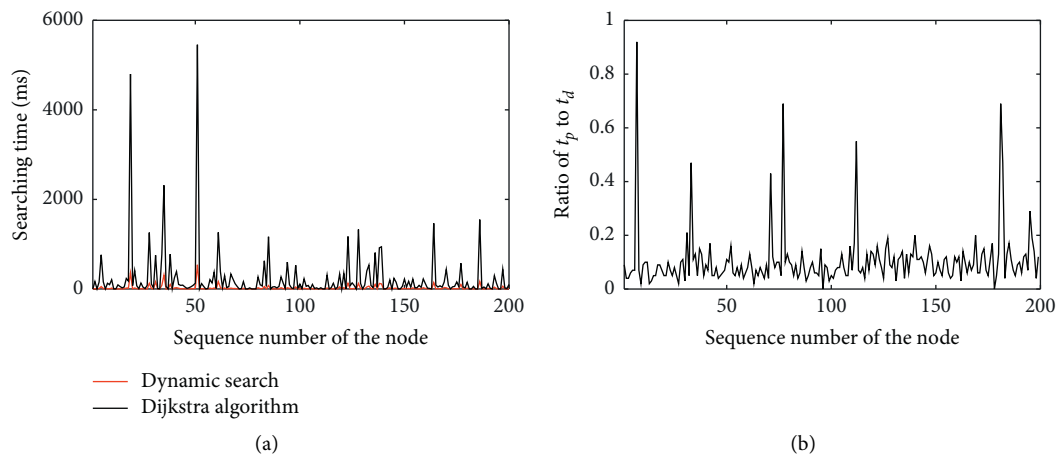


FIGURE 5: The performance comparison between the proposed dynamic algorithm and the Dijkstra algorithm: (a) comparison of the searching time; (b) ratio of t_p to t_d .

value of 0 since they have no node pass through the fire area. At node N_{51} , there are 256 paths that need to be modified and the ratio is 0.1. At node N_{35} , there are 178 paths that need to be modified and the ratio is 0.05. It is efficacious to use the proposed dynamic algorithm as there are many nodes in the SNode.

5. Conclusions

The prior evacuation paths are inefficient when the situations are different from the prospective environments. It takes a long time to search for the shortest paths from each node to the exits, and it is unsuitable for online searching. The proposed dynamic algorithm adjusts the prior paths that pass through the fire areas according to the situations. Because the proposed method takes advantages of the prior evacuation path network, it visits the nodes around the fire area to get the path information instead of searching each node of the model. It modifies the paths that pass through the fire areas in a short time, and each path is the shortest one from itself to the exit. The proposed dynamic algorithm

is suitable for the online evacuation path optimization because of its fast convergence and global searching.

Data Availability

The data used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

- [1] S. Aalami and L. Kattan, "Fairness and efficiency in pedestrian emergency evacuation: modeling and simulation," *Safety Science*, vol. 121, pp. 373–384, 2020.
- [2] 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.
- [3] R. E. Meouche, M. Abunemeh, I. Hijaze, A. Mebarki, and I. Shahrour, "Developing optimal paths for evacuating risky

- construction sites,” *Journal of Construction Engineering and Management*, vol. 144, no. 2, Article ID 04017099, 2018.
- [4] J. M. Czerniak, H. Zarzycki, Ł. Apiecionek, W. Palczewski, and P. Kardasz, “A cellular automata-based simulation tool for real fire accident prevention,” *Mathematical Problems in Engineering*, vol. 2018, Article ID 3058241, 12 pages, 2018.
 - [5] H. Mi, Y. Liu, W. Wang, and G. Xiao, “An integrated method for fire risk assessment in residential buildings,” *Mathematical Problems in Engineering*, vol. 2020, Article ID 9392467, 14 pages, 2020.
 - [6] I. Bakas, K. Georgiadis-Filikas, and K. J. Kontoleon, “Treasures gutted by fire. Fire safety design awareness as a consequence of historic building accidents and disasters,” *IOP Conference Series. Earth and Environmental Science*, vol. 410, no. 1, Article ID 12113, 2020.
 - [7] Z. Dou, A. Mebarki, Y. Cheng et al., “Review on the emergency evacuation in chemicals-concentrated areas,” *Journal of Loss Prevention in the Process Industries*, vol. 60, pp. 35–45, 2019.
 - [8] R. R. M. Gershon, “The world trade center evacuation study: factors associated with initiation and length of time for evacuation,” *Fire and Materials*, vol. 36, no. 5-6, pp. 481–500, 2012.
 - [9] C. Minji, “Effect of dynamic emergency cues on fire evacuation performance in public buildings,” *Journal of Infrastructure Systems*, vol. 24, no. 4, 2018.
 - [10] P. Chen and F. Feng, “A fast flow control algorithm for real-time emergency evacuation in large indoor areas,” *Fire Safety Journal*, vol. 44, no. 5, pp. 732–740, 2019.
 - [11] F. Mirahadi and B. Y. McCabe, “EvacuSafe: a real-time model for building evacuation based on Dijkstra’s algorithm,” *Journal of Building Engineering*, vol. 34, Article ID 101687, 2020.
 - [12] W. Y. Lin and P. H. Lin, “Intelligent generation of indoor topology (i-GIT) for human indoor pathfinding based on IFC models and 3D GIS technology,” *Automation in Construction*, vol. 94, pp. 340–359, 2018.
 - [13] 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, 2015.
 - [14] Y. Zhao, H. Liu, and K. Gao, “An evacuation simulation method based on an improved artificial bee colony algorithm and a social force model,” *Applied Intelligence*, vol. 51, no. 1, pp. 100–123, 2021.
 - [15] L. Liu, H. Zhang, J. Xie, and Q. Zhao, “Dynamic evacuation planning on cruise ships based on an improved ant colony system (IACS),” *Journal of Marine Science and Engineering*, vol. 9, no. 2, p. 220, 2021.
 - [16] F. Li, “Modelling multi-exit large-venue pedestrian evacuation with dual-strategy adaptive particle swarm optimization,” *IEEE Access*, vol. 8, pp. 114554–114569, 2021.
 - [17] Y. Zhang, D. Zhang, and J. Jin, “Evacuation path optimization algorithm for inland river passenger ship in emergency situation,” *Journal of Coastal Research*, vol. 83, p. 256, 2018.
 - [18] Y.-J. Zheng, H.-F. Ling, J.-Y. Xue, and S.-Y. Chen, “Population classification in fire evacuation: a multiobjective particle swarm optimization approach,” *IEEE Transactions on Evolutionary Computation*, vol. 18, no. 1, pp. 70–81, 2014.
 - [19] N. Khamis, H. Selamat, F. S. Ismail, O. F. Lutfy, M. F. Haniff, and I. N. A. M. Nordin, “Optimized exit door locations for a safer emergency evacuation using crowd evacuation model and artificial bee colony optimization,” *Chaos, Solitons & Fractals*, vol. 131, Article ID 109505, 2020.
 - [20] 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.
 - [21] C. Wang, L. C. Wood, H. Li, Z. Aw, and A. Keshavarzsaleh, “Applied artificial bee colony optimization algorithm in fire evacuation routing system,” *Journal of Applied Mathematics*, vol. 2018, Article ID 7962952, 17 pages, 2018.
 - [22] L. Yapeng, C. Wei, and A. K. Austin, “Design of level of service on facilities for crowd evacuation using genetic algorithm optimization,” *Safety Science*, vol. 120, pp. 237–247, 2019.
 - [23] Z. Yu, D. Li, S. Zhu, W. Luo, Y. Hu, and L. Yuan, “Multisource multisink optimal evacuation routing with dynamic network changes: a geometric algebra approach,” *Mathematical Methods in the Applied Sciences*, vol. 41, no. 11, pp. 4179–4194, 2018.
 - [24] Y. Amirgaliyev, R. Yunussov, and O. Mamyrbayev, “Optimization of people evacuation plans on the basis of wireless sensor networks,” *Open Engineering*, vol. 6, no. 1, pp. 206–213, 2016.
 - [25] R. Löhner, E. Haug, C. Zinggerling, and E. Oñate, “Real-time micro-modelling of city evacuations,” *Computational Particle Mechanics*, vol. 5, no. 1, pp. 71–86, 2018.
 - [26] J.-S. Chou, M.-Y. Cheng, Y.-M. Hsieh, I.-T. Yang, and H.-T. Hsu, “Optimal path planning in real time for dynamic building fire rescue operations using wireless sensors and visual guidance,” *Automation in Construction*, vol. 99, pp. 1–17, 2019.
 - [27] V. Karthik and S. Suja, “Optimized multiple existence for pedestrian evacuation using geographic map-based path discovery,” *Cluster Computing*, vol. 22, no. S5, pp. 11227–11236, 2019.
 - [28] Sunita and D. Garg, “A retroactive approach for dynamic shortest path problem,” *National Academy Science Letters*, vol. 42, no. 1, pp. 25–32, 2019.
 - [29] A. Sedeño-noda and M. Colebrook, “A biobjective Dijkstra algorithm,” *European Journal of Operational Research*, vol. 276, no. 1, pp. 106–118, 2019.
 - [30] Sunita and D. Garg, “Dynamizing Dijkstra: a solution to dynamic shortest path problem through retroactive priority queue,” *Journal of King Saud University—Computer and Information Sciences*, vol. 33, no. 3, pp. 364–373, 2021.
 - [31] M. Choi and S. Chi, “Optimal route selection model for fire evacuations based on hazard prediction data,” *Simulation Modelling Practice and Theory*, vol. 94, pp. 321–333, 2019.
 - [32] A. Veeraswamy, E. R. Galea, L. Filippidis et al., “The simulation of urban-scale evacuation scenarios with application to the Swinley forest fire,” *Safety Science*, vol. 102, pp. 178–193, 2018.
 - [33] Y. Kazuhiro, S. Yuusuke, and N. Shinnosuke, “Simulation of tunnel fire for evacuation safety assessment,” *Safety*, vol. 4, no. 2, p. 12, 2018.
 - [34] X. Zhao, R. Lovreglio, and D. Nilsson, “Modelling and interpreting pre-evacuation decision-making using machine learning,” *Automation in Construction*, vol. 113, Article ID 103140, 2020.
 - [35] Y. Ma, K. Liu, M. Chen et al., “ANT: deadline-aware adaptive emergency navigation strategy for dynamic hazardous ship evacuation with wireless sensor networks,” *IEEE Access*, vol. 8, pp. 135758–135769, 2020.
 - [36] A. Darvishan and G. J. Lim, “Dynamic network flow optimization for real-time evacuation reroute planning under multiple road disruptions,” *Reliability Engineering & System Safety*, vol. 214, Article ID 107644, 2021.

- [37] D. Hartama, H. Mawengkang, M. Zarlis et al., "Evacuation planning for disaster management by using the relaxation based algorithm and route choice model," *IFIP Advances in Information and Communication Technology*, vol. 516, pp. 136–144, 2019.
- [38] I. A. Ndiaye, E. Neron, and A. Jouglet, "Macroscopic evacuation plans for natural disasters," *OR Spectrum*, vol. 39, no. 1, pp. 231–272, 2017.
- [39] Y. Peng, S.-W. Li, and Z.-Z. Hu, "A self-learning dynamic path planning method for evacuation in large public buildings based on neural networks," *Neurocomputing*, vol. 365, pp. 71–85, 2019.
- [40] K. Tkachuk, X. Song, and I. Maltseva, "Application of artificial neural networks for agent-based simulation of emergency evacuation from buildings for various purpose," *IOP Conference Series, Materials Science and Engineering*, vol. 365, no. 4, Article ID 42064, 2018.
- [41] S. M. Lo, M. Liu, P. H. Zhang, and R. K. K. Yuen, "An artificial neural-network based predictive model for pre-evacuation human response in domestic building fire," *Fire Technology*, vol. 45, no. 4, pp. 431–449, 2009.
- [42] W. Liu, "A dynamic evacuation algorithm based on CCRP under fire conditions," *China Work Safety Science and Technology*, vol. 16, no. 5, pp. 32–37, 2020.
- [43] J. Radianti, O.-C. Granmo, P. Sarshar, M. Goodwin, J. Dugdale, and J. J. Gonzalez, "A spatio-temporal probabilistic model of hazard- and crowd dynamics for evacuation planning in disasters," *Applied Intelligence*, vol. 42, no. 1, pp. 3–23, 2015.
- [44] M. Saeed Osman and B. Ram, "Distributed scheduling approach for dynamic evacuation networks," *Mathematical and Computer Modelling of Dynamical Systems*, vol. 23, no. 6, pp. 554–569, 2017.
- [45] T. N. Dhamala, U. Pyakurel, and S. Dempe, "A critical survey on the network optimization algorithms for evacuation planning problems," *International Journal of Operations Research*, vol. 15, no. 3, pp. 101–133, 2018.
- [46] J. Qin, C. Liu, and Q. Huang, "Simulation on fire emergency evacuation in special subway station based on Pathfinder," *Case Studies in Thermal Engineering*, vol. 21, Article ID 100677, 2020.
- [47] D. Yin, S. Wang, and Y. Ouyang, "ViCTS: a novel network partition algorithm for scalable agent-based modeling of mass evacuation," *Computers, Environment and Urban Systems*, vol. 80, Article ID 101452, 2020.