

# An Algorithm for Calculating Simple Evacuation Routes in Evacuation Guidance Systems

Koichi Asakura and Toyohide Watanabe

**Abstract** This paper proposes an algorithm for calculating simple evacuation routes in evacuation guidance systems. In order to ensure that evacuees move to the shelters safely in disaster situations, our algorithm produces the simple routes in which the minimum number of turns is included. Evacuees can move to the shelters by following simple routes with low risk of making a mistake. For calculating simple routes, a road network is transformed so that an edge in the road network is converted into a vertex and a vertex is converted into several edges. Experimental results show that the length of the produced simple routes is not so different from that of the shortest routes.

**Keywords** Evacuation guidance systems · Simple routes · Disaster simulation · Evacuee agent

## 1 Introduction

In disaster situations such as major earthquakes, it is extremely important for people in the affected areas to safely and correctly follow evacuation routes to the nearest shelters. Recent developments with smartphones and their applications mean we can now use many kinds of map applications, anywhere and anytime. With online map applications such as Google Maps [1], we can identify the best routes to a destination while considering the current situations in the areas. Also, offline map applications such as MapsWithMe [2] enables us to reference map data without a network

---

K. Asakura (✉)

Department of Information Systems, School of Informatics, Daido University,  
10-3 Takiharu-cho, Minami-ku, Nagoya 457-8530, Japan  
e-mail: asakura@daido-it.ac.jp

T. Watanabe

Nagoya Industrial Science Research Institute, 1-13 Yotsuya-dori, Chikusa-ku,  
Nagoya 464-0819, Japan  
e-mail: watanabe@nagoya-u.jp

connection. These map applications are enormously useful in terms of helping people in the disaster areas to move to the nearest shelters for evacuation.

However, it is possible that the usual evacuation routes retrieved by the above map applications may not be feasible for evacuation since roads may be impassable due to rubble from collapsed buildings, the spread of fire, and so on in the disaster areas. In response to this issue, we have already proposed a construction system for navigational maps [3]. In this system, we adopt a mechanism based on ant colony systems in order to take recentness of information into account [4, 5].

In addition, in such affected areas, special considerations for selecting evacuation routes may be required. For example, in confusing areas, evacuation routes must be simple enough to lower the risk of evacuees making a navigational mistake, rather than just being the shortest. People tends to clump together and move to the shelters as a group in evacuation activities, and in order to avoid being separated from the group, they should select simple routes rather than short but difficult ones.

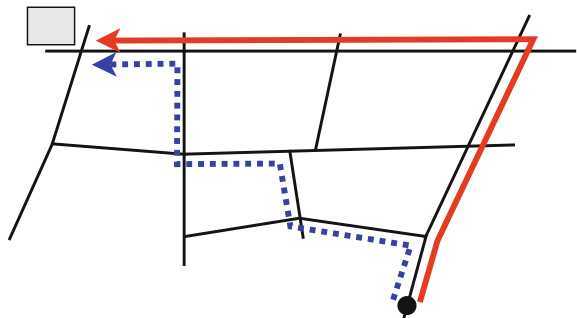
In this paper, we propose an algorithm for calculating simple evacuation routes for evacuation guidance systems. Here, we define a simple route as the route in which the number of turns at intersections is the fewest. In our algorithm, a road network in the map is transformed into another network and then Dijkstra's algorithm is applied to the transformed network.

The rest of this paper is organized as follows. Section 2 describes the main issue that we tackle in this paper, namely, providing the simplest routes. Section 3 gives an algorithm for calculating simple routes and Sect. 4 evaluates the proposed algorithm with simulation experiments. We conclude in Sect. 5 concludes with a brief summary and details on our future work.

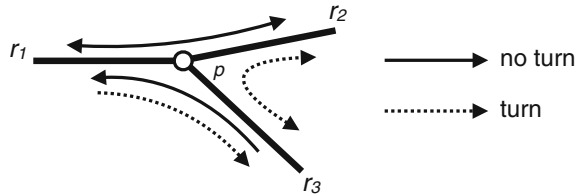
## 2 Problem Description

Our proposed evacuation guidance systems helps evacuees in disaster areas to take safe and secure routes to the shelters. For this purpose, we define a simple route as the route to the shelter along which evacuees have to turn at intersections at the minimum number of times. Figure 1 shows an example of the shortest route and the

**Fig. 1** The shortest route and the simple route



**Fig. 2** Asymmetry of turn at an intersection



simple route to the shelter. In this figure, a shelter is denoted as a square, and an evacuee is denoted as a bullet. Line segments show the road network. The dashed blue arrow shows the shortest route from the position of the evacuee to the shelter. Conventional route calculating algorithms such as Dijkstra’s algorithm, the Bellman-Ford algorithm, and the A\* algorithm [6], can produce the shortest route easily, but as shown in the figure, the evacuee has to turn at the intersections many times along the shortest route. In contrast, the red arrow shows the simple route. We can observe that this route has only one turn. Although the length of the simple route is slightly longer than that of the shortest one, the simple routes enables evacuees to move to the shelter easily.

In order to calculate the simple route in a road network, we have to consider asymmetry of turn at intersections. In other words, it depends on the movement direction whether an evacuee has to turn at an intersection or not even if the evacuee passes through the same two road segments at the same intersection. Figure 2 shows an example of this situation. In this figure, the road segments  $r_1$ ,  $r_2$  and  $r_3$  connect at the intersection  $p$ . When an evacuee passes through at  $p$  from  $r_1$  to  $r_2$  and from  $r_2$  to  $r_1$ , the evacuee moves along the road without turning. Furthermore, when the evacuee takes routes from  $r_2$  to  $r_3$  and from  $r_3$  to  $r_2$ , one turn is counted at these movements. However, there is an asymmetrical relationship between the route from  $r_1$  to  $r_3$  and the route from  $r_3$  and  $r_1$ . When evacuees move from  $r_1$  to  $r_3$ , they have to turn right at the intersection  $p$ . However, no turns are required for the movement from  $r_3$  to  $r_1$  because evacuees just follow the road and thus the navigation has no difficulties.

In order to calculate simple routes, this issue has to be resolved. In the next section, we propose a transformed road network for handling asymmetry of turn.

### 3 Algorithm

In this section, we propose an algorithm for calculating simple routes for easy evacuation in disaster areas. For the sake of resolving the issue of asymmetry discussed above, road networks are transformed into another type of network. Section 3.1 discusses a transformed road networks and Sect. 3.2 presents the algorithm for calculating simple evacuation routes.

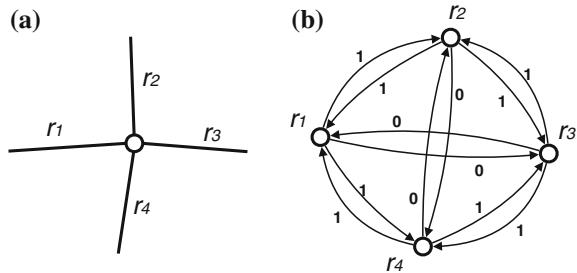
### 3.1 Transformation of Road Networks

As described in Sect. 2, in order to calculate simple routes for evacuation, we have to deal with asymmetry of turn at intersections. For this issue, we propose using transformed road networks. In original road networks, road segments are denoted as edges and connecting points among road segments are denoted as vertices. In transformed road networks, the edges and the vertices in the original road network are switched into vertices and edges, respectively. Thus, one connecting point in the original road network is transformed into several edges between vertices. Additionally, these edges have a cost 0 or 1, where cost 0 means that people can walk along two road segments without turning, and cost 1 means that they have to turn at the intersection between two road segments. The transformed road networks are directed networks, namely, the edges in the transformed road network are directed edges.

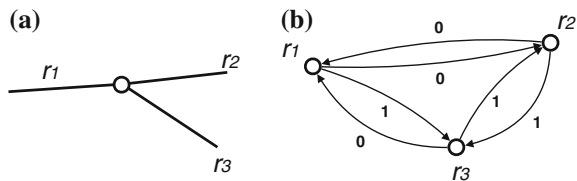
Figure 3 shows an example of transformed road networks. An original road network is shown in Fig. 3a. In this road network, four road segments are connected at one intersection. Since this intersection is a general four-way intersection, there are no turns between  $r_1$  and  $r_3$ , and between  $r_2$  and  $r_4$ . Also, there are turns between  $r_1$  and  $r_2$ , between  $r_1$  and  $r_4$ , and so on. The transformed road network is shown in Fig. 3b. In this network, one intersection is transformed into 12 directed edges each with a cost 0 or 1. From this transformed network, we can assume that it costs 1 from  $r_3$  to  $r_4$ , namely, we have to turn at the intersection from  $r_3$  to  $r_4$ .

An example of asymmetry of turn at the intersection is given in Fig. 4. In Fig. 4a, people turns at the intersection from  $r_1$  to  $r_3$ , but no turns are required from  $r_3$  to  $r_1$ . As shown in Fig. 4b, our transformed road network can handle this situation since the transformed road network is a directed network.

**Fig. 3** Example of a transformed road network.  
**a** original road network  
**b** transformed network



**Fig. 4** A transformed road network for asymmetry of turn.  
**a** original road network  
**b** transformed network



### 3.2 Calculation of Simple Routes

After constructing the transformed road network, we can easily calculate a simple route to the shelter by using an algorithm for calculating the shortest path, such as Dijkstra's algorithm, the Bellman-Ford algorithm, or the A\* algorithm [6]. The calculated cost represents the minimum number of turns from each road segment to the shelter.

## 4 Evaluation

In this section, we evaluate the proposed algorithm. Section 4.1 compares the simple route with the shortest route with a practical road network. Section 4.2 analyzes the time complexity of the proposed algorithm.

### 4.1 Experimental Results

For experimental evaluation, we used a map of the area surrounding our university (Fig. 5). This map covers an area 2.0 km wide and 1.5 km high. There is a residential



Fig. 5 A practical road network for experimental evaluation

area at the bottom right of the map. Thus, we can observe a road network of a fine-grained checkerboard pattern. We provide one shelter on the map and calculate shortest routes and simple routes from each connecting point on the map. We then compare the number of turns in the routes and the length of the routes. We have 591 connecting points on the map.

Figures 6 and 7 show typical comparative results. In these figures, blue lines denote shortest routes and red lines denote simple routes. In Fig. 6a, the shortest route forms a zigzag pattern, since it passes through the residential area at the bottom right of the map. In contrast, with the calculated simple route in Fig. 6b, evacuees can pass through the residential area without turning. Also, we can observe from Fig. 7 that only one turn is required for evacuation with the calculated simple route and that the length of the two routes is almost the same.

Table 1 summarizes the experimental results statistically. The average values of the number of turns in the routes and the length of the routes are shown. From these results, it is clear that our proposed algorithm can ably calculate the simple routes. Also, we can observe that the average length of the simple routes is not so different from that of the shortest routes. This demonstrates that the proposed algorithm can produce better evacuation routes.

## 4.2 Computational Complexity

In the proposed algorithm, in order to calculate simple routes, a road network is reformed into a transformed road network and the shortest routes are then calculated in the transformed network. Here, a road network is denoted as  $G = (V, E)$ , where  $V$  is a set of connecting points and  $E$  is a set of road segments. In the same way, a transformed road network is denoted as  $G' = (V', E')$ .

As described in Sect. 3.1, because the vertices in the transformed road network correspond to the edges in the original road network, the following equation is established:

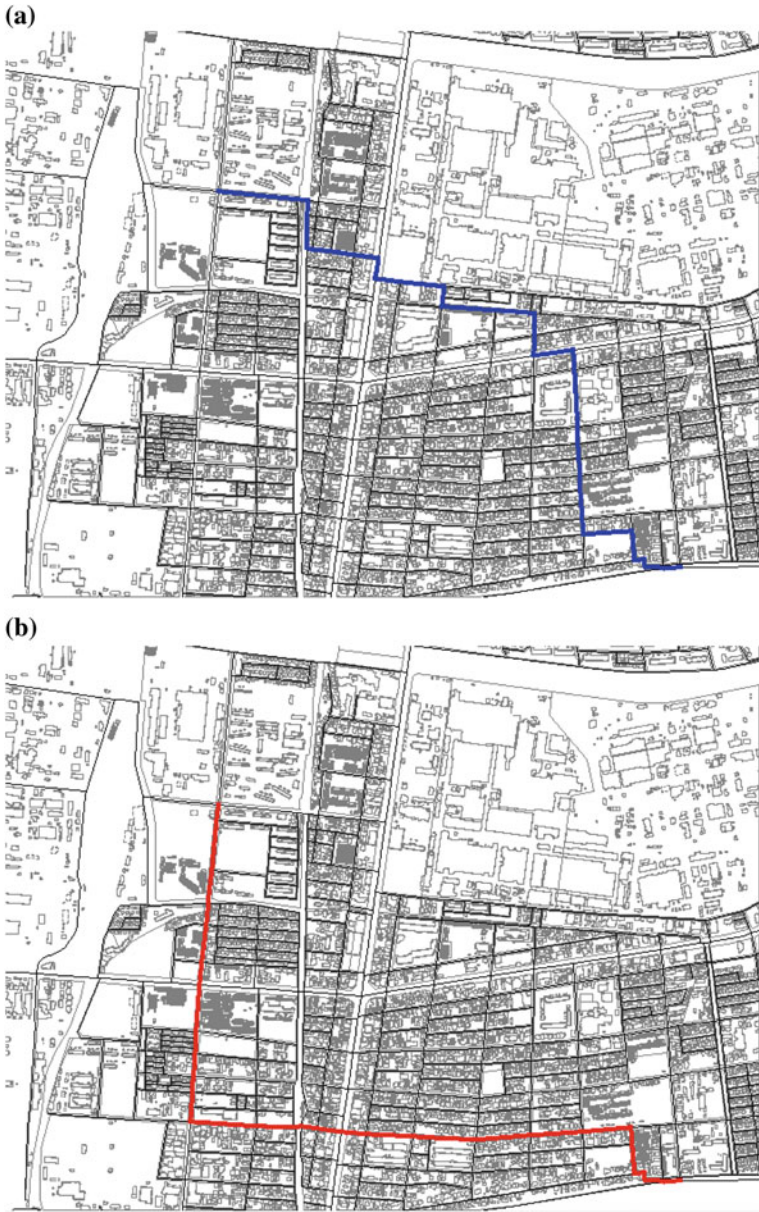
$$|V'| = |E|, \quad (1)$$

where  $|\cdot|$  expresses the number of elements of the set.

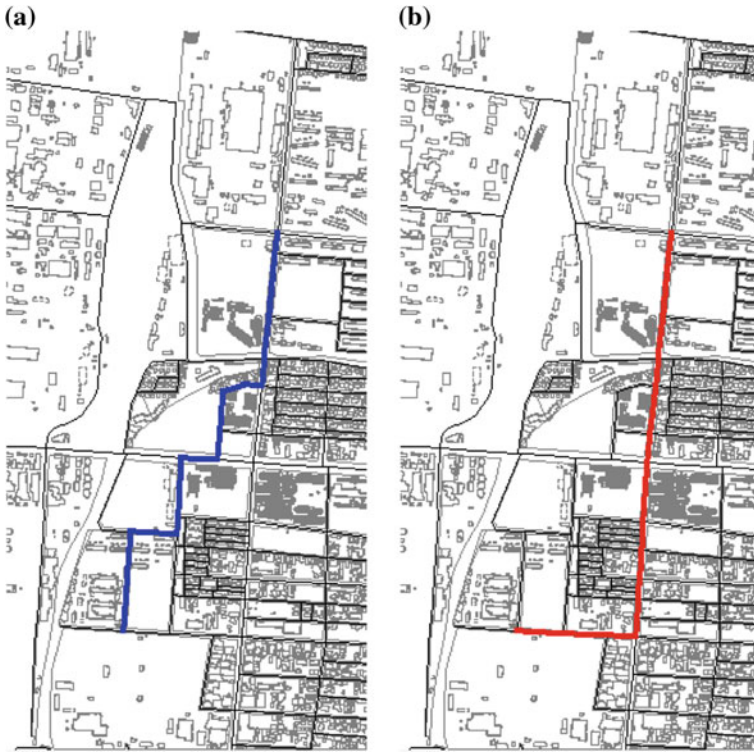
When  $n$  road segments are connected at one intersection in the original road network, the number of edges for the intersection in the transformed road network is described as  ${}_n P_2 = n \cdot (n - 1)$ . Thus, the number of edges in the transformed road network can be described as

$$|E'| = \sum_{v \in V} (deg(v) \cdot (deg(v) - 1)), \quad (2)$$

where  $deg(v)$  denotes the degree of the vertex  $v$  [7]. Here, as an approximate value for  $deg(v)$ , we adopt the average degree:



**Fig. 6** Experimental result 1. **a** The shortest route. Turns: 14. Length: 2142 m **b** The shortest route. Turns: 5. Length: 2271 m



**Fig. 7** Experimental result 2. **a** The shortest route. Turns: 5. Length: 1077 m **b** The shortest route. Turns: 1. Length: 1086 m

**Table 1** Statistical results for experiments

	Turns	Length [m]
Shortest routes	2.53	1064.40
Simple routes	1.38 (54.5 %)	1152.90 (108.3 %)

$$deg(v) = \frac{2|E|}{|V|}. \tag{3}$$

Thus, the number of edges in the transformed road network is approximated as

$$|E'| = \sum_{v \in V} \frac{2|E|}{|V|} \cdot \left( \frac{2|E|}{|V|} - 1 \right) = 2|E| \cdot \left( \frac{2|E|}{|V|} - 1 \right). \tag{4}$$

For  $G = (V, E)$ , the computational complexity of Dijkstra's algorithm is denoted as

$$O(|E| + |V| \log |V|). \tag{5}$$



Thus, the computational complexity for calculating simple routes is denoted as

$$\begin{aligned} O(|E'| + |V'| \log |V'|) &= O\left(\left(2|E| \cdot \frac{2|E|}{|V|} - 1\right) + |E| \log |E|\right) \\ &= O\left(\frac{|E|^2}{|V|} + |E| \log |E|\right). \end{aligned} \quad (6)$$

## 5 Conclusion

In this paper, we have proposed an algorithm for calculating simple evacuation routes for evacuation guidance systems in disaster areas. With simple routes, evacuees can move to shelters easily without any mistakes because they only have to turn at intersections the minimum number of times. Experimental results show that our proposed algorithm can produce simple routes and that the length of the simple routes is not so different from that of the shortest routes.

In our future work, we will develop a route calculation algorithm that takes the trade-off between length of the route and simplicity of the route into account. This issue is a kind of optimization problem with multi-variable functions.

## References

1. Google Inc.: Google Maps. <https://maps.google.com/>
2. My.com: MAPS.ME. <http://maps.me/>
3. Asakura, K., Watanabe, T.: Construction of navigational maps for evacuees in disaster areas based on ant colony systems. *Int. J. Knowl. Web Intell.* **4**(4), 300–313 (2013)
4. Dorigo, M., Stützle, T.: *Ant Colony Optimization*. Bradford Company (2004)
5. Blum, C.: Ant colony optimization: introduction and recent trends. *Phys. Life Rev.* **2**(4), 353–373 (2005)
6. Cormen, T.H., Leiserson, C.E., Rivest, R.L., Stein, C.: *Introduction to Algorithms*. MIT Press (2009)
7. Diestel, R.: *Graph Theory*. Springer (2010)