

Simulation of Optimized Evacuation Processes based on Sub-Goal CA Model

Zhiwen Hu^{1,2}, Huaqing Mao^{3,*} and Li Zhu³

¹ School of New Media, Zhejiang University of Media and Communications, Hangzhou 310018, P.R. China

² Department of Mechanical Engineering, University of Victoria, Victoria V8W 3P6, Canada

³ Oujiang College, Wenzhou University Chashan University Town Wenzhou, Zhejiang 325035, P.R. China
sunneyhu@me.com; mr.maohuaqing@qq.com; yeah_1397118@hotmail.com

Abstract

With the full swing of computer technology, evacuation models are constructed and simulated via computers to study the various phenomena and behaviors during the evacuation process. They open unique potential opportunity to evaluate the architectural design and provide the decision support for the emergency evacuation plans. This paper details a capacity constrained route planner (CCRP) algorithm, which is used to optimize evacuation path, and the distance map based on sub-goal cellular automata (CA) is also proposed to simulate the evacuation scenarios. Two evacuation scenarios are further discussed, and the experimental results strongly suggested that the sub-goal CA model could simulate the evacuation scenarios and evaluate the emergency evacuation plans effectively and finally help us penetrate to the essence of reality.

Keywords: Evacuation Model; Capacity Constrained Route Planner; Cellular Automata; Traffic Flow Model; Optimum Solution

1. Introduction

With the rapid development of social economy and acceleration of the process of urbanization and industrialization, huge architectures like airports, hotels, stations, markets, and other high buildings have mushroomed all over the world. How to direct the herd out of those places in emergency, is crying for answer[1,2,3,4]. The key to address these challenges is the ability to account the optimum emergency response in building evacuation as the most important part in evacuation plan. Traditionally, evacuation environment are described as spatial network, and crowds are described as stream. Lots of works have been done by using spatial network and stream to construct the evacuation model, Hamacher and Tjandra summarized the linear programming (LP) to solve the cheapest evacuation route[5].

In recent years, there has been an ever-increasing interest in the use of cellular automata (CA) model [6] in traffic flow model [7,8,9], which provided the basic foundation of crowd flow. To address the efficiency and validation problems in evacuation model, this paper proposes a CA model based on sub-goal model, which is simplified and improved from the traffic flow model (TFM), and capacity constrained route planner (CCRP) algorithm [10] is introduced in this model to solve the evacuation path. The evacuation scenarios are simulated by the distance map, which is obtained by algorithm very similar to the Dijkstra algorithm.¹

¹ Huaqing Mao is the corresponding author.

The rest of the paper is organized as follows: In Section 2 the CCRP algorithms are described. The definition of CA based on sub-goal and the distance map are proposed in Section 3. The experimental setup and the results of performance comparison are reported in Section 4. The final section summarizes the result.

2. Problem Statement and Preliminaries

2.1 Algorithm Initializations

The description of CCRP algorithm Initializations are divided into four parts as following:

Input:

(1) Spatial evacuation network, $G(N,E)$: G is evacuation network, N is the set of points, and E is the set of edges.

For any node $n \in N$ has two properties:

Maximum_Node_Capacity(n): The max capacity of node, positive integers.

Initial_Node_Occupancy(n): The initial pedestrian's number of node, positive integers.

For any edge $e \in E$ has two properties:

Maximum_Edge_Capacity(e): The max capacity of edge, positive integers.

Travel_time(e): The time of passing through the edge, positive integers.

(2) S : Source nodes set, $S \in N$.

(3) D : Target nodes set, $D \in N$.

Output:

The output of algorithm is an evacuation plan, and it includes a set of initial position to target position path and evacuation path of each pedestrian. The number of pedestrian should not exceed the capacity of nodes and edges at the same time.

Goal:

(1) Achieve the shortest evacuation time.

(2) Calculate the evacuation plan by using less time.

Constraints:

(1) Evacuation plan should follow the rule of first-in first-out (FIFO).

(2) The simulation should reflect the influence of jams.

(3) The time of solving process and space complexity should be actual operation.

2.2 Algorithm Description

The output of algorithm is an evacuation plan, and it includes a set of initial position to target position path and evacuation path of each pedestrian[11,12]. The number of pedestrian should not exceed the capacity of nodes and edges at the same time.

CCRP is a heuristic optimization algorithm, which can calculate the second best evacuation plan result in the complexity evacuation network with large amount

evacuate pedestrians[13,14,15]. According to the algorithm initializations, the algorithm process is described as follow:

- (1) Evacuation network preprocess: Add a super source node S_0 to the evacuation network, and add the edges that connect to the S_0 , assign the value to the two properties of edge:

$$\begin{aligned} \text{Maximum_Node_Capacity}(e) &= \infty, \\ \text{Travel_time}(e) &= 0. \end{aligned}$$

- (2) Loop process: for and source node s ($s \in S$) with pedestrian in it, then loop start.

- (a) Take source node S_0 as original node, and calculate the shortest path from S_0 to all target nodes D by using Dijkstra algorithm. Let $R \langle n_0, n_1, \dots, n_k \rangle$ indicates the shortest path from source node s ($s \in S$) to target node d ($d \in D$), and let $\langle t_0, t_1, \dots, t_{k-1} \rangle$ indicates time plan of R ($n_i (i=0, 1, \dots, k)$ means the nodes of path, and $t_i (i=0, 1, \dots, k-1)$ means departure time from the node).

Then we have:

$$\begin{aligned} n_0 = s, n_k = d, \text{Available_Edge_Capacity}(e_{n_i n_{i+1}}, t_i) &> 0, \forall i \in \{0, 1, \dots, k-1\}; \\ \text{Available_Node_Capacity}(n_{i+1}, t_i + \text{Travel_Time}(e_{n_i n_{i+1}})) &> 0, \forall i \in \{0, 1, \dots, k-1\} \end{aligned}$$

- (b) Calculate the pedestrian's number from the node s by:

$$\text{flow} = \min \left(\begin{array}{l} \text{The number of pedestrian in node } s, \\ \text{Available_Edge_Capacity}(e_{n_i n_{i+1}}, t_i), \\ \text{Available_Node_Capacity}(n_{i+1}, t_i + \text{Travel_time}(e_{n_i n_{i+1}})) \end{array} \right), \forall i \in \{0, 1, \dots, k-1\}$$

- (c) According the person number recalculates the surplus capacity of nodes and edges by:

- i. for $i = 0$ to $k-1$ do{
- ii. $\text{Available_Edge_Capacity} - \text{flow}(e_{n_i n_{i+1}}, t_i)$
- iii. $\text{Available_Node_Capacity} - \text{flow}(n_{i+1}, t_i + \text{Travel_time}(e_{n_i n_{i+1}}))$
- iv. }

- (d) Loop end.

- (3) Output the evacuation plan.

2.3 Algorithm Analysis

In Section 2.2, super source node S_0 is introduced to the evacuation network preprocesses, and it is connected to the other source node by edge, which has infinitely capacity and 0 travel time. The fastest path of any source nodes to any target nodes is determined by executing fastest path algorithm start with S_0 only once. The comparison between CCRP and the linear programming (LP) algorithm is shown in table 1.

Table 1. Algorithm Comparison

Algorithm	Time Complexity	Solution type
CCRP	$O(p \cdot n \log n)$	suboptimum
LP	At least $O((T \cdot n)6)$	solution optimum solution

In table 1, n is the number of nodes, p is the number of evacuate people, T is the upper limit of evacuation time. The result shows that the efficiency to get near-optimal result of CCRP is far superior to the linear programming algorithm, especially in the complex environmental inner buildings with large number of evacuate people.

3. CA Model based on Sub-Goal

3.1 Model Description

In this model, escape room is equally divided into $W \times H$ square grid, each grid present a Cellular, and W, H means the size of X, Y axis[16]. In a same time step, each cellular has two states: empty or occupy. In most cases, $0.4 \times 0.4 \text{m}^2$ grid has highly efficient computation [17], thus we use it as a default grid in the evacuation model.

In our evacuation model, position of pedestrian is synchronous updated, and the new status of cellar is determined by the neighbor which is defined as the all cellars within 2 steps around the current cellar. In each time step, person can only move 1 step or stay in current position. The time step is determined by pedestrian moving speed. It's been observed that the average speed is about 1.3 m/s in normal case, for the tense situation speed will be 1.8 m/s, which the associated time step will be $0.4/1.8 \approx 0.22\text{s}$. In the evacuation process, the moving principles are as following:

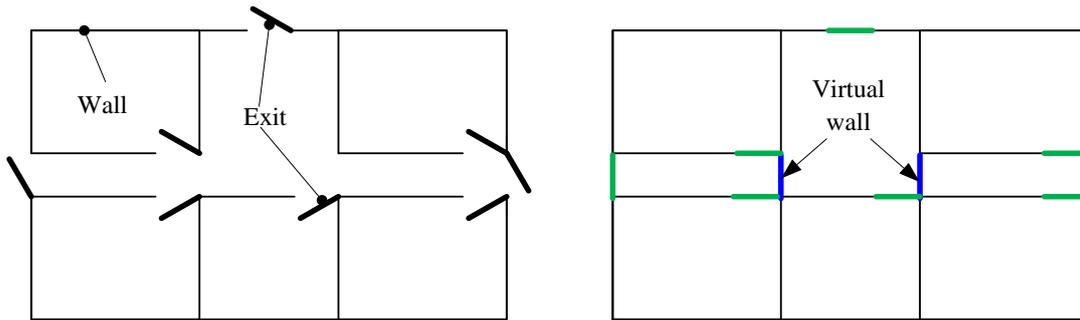
- (1) Try to moving towards the nearest exit.
- (2) Try to moving towards the lower crowd density.
- (3) Steer away from the obstructions.

3.2 Sub-Goal Definition

Inside the building, the visible range that pedestrian can see is limited in the small space; pedestrian don't know the situation outside the space. When the fire broke out, the first step of evacuation is leaving the current position, and looking for the evacuation path of next step. Every moment, pedestrian are in the particular space, before they leave the space, all the movement are relevant to the information of the space. According the floor plan of the building, it can be divided into multiple small and simple subspaces which are completely sealed and consisted of walls and exits. For those public spaces like corridor, stair, lobby, we can seal them with virtual wall and generate the subspace. The connection that joins the subspace is defined as exit of subspace and considered as sub-goal of pedestrian inside the current subspace.

Before the evacuation, base the globe evacuation environment and crowd distribution circumstance, system choose the exit and best evacuation path for every evacuate pedestrian by using the CCRP algorithm. After evacuation begins, pedestrian in the subspace will move to the next sub-goal until they reach the final exit.

The example of the sub-goal definitions which are marked with green and blue color is shown in Figure 1. The building floor plan has 5 rooms and 1 corridor, each room has a door connect to corridor, and the corridor connects to 2 final exits. According the sub-goal definition method, each room is defined as a subspace, the door of the room is defined as sub-goal, the corridor is divided into 3 dependence subspace by the blue virtual walls and the connection of subspace is defined as their sub-goal.



**Figure 1. Example of Sub-Goal Definition
(Green and Blue Segments is Sub-Goal)**

3.3 Distance Map Generation

Distance map is a grid map covering the building floor plan, and a value is stored in each grid, the shortest distance between the current grid and the sub-goal of current space is recorded in the value. In the model we proposed, the distance unit is time step. From the distance map definition, it will be seen that each distance map should correspond to a sub-goal, but each sub-goal may correspond to two distance maps because every exit (including virtual exit) should connect two subspaces. The value stored in grid is not only representing the linear distance between the current grid and the current sub-goal, but also representing the influence of static obstruction such as walls.

The purpose of distance map is to ensure that people move to the grid which is near the current sub-goal during the evacuation, and the other purpose is to guarantee the people avoiding the static obstruction without extra obstruction detection which is promoting efficiency of algorithm.

The steps of generation are as following:

- (1) Identifying the border of subspace covered by distance map (X_{max} , Y_{min} , X_{max} , Y_{max}), and identifies the grid coordinates of the target exit. In the distance map, each grid stores two variables: ($minDistance$, $available$). $minDistance$ indicate the shortest distance between the current grid and the current sub-goal, $available$ indicate whether the current grid is null. It will have the value 1 when the grid is null, otherwise it will have the value 0. Before the distance map generation, all the $minDistance$ of the grid are assigned an infinite number (e.g. 99999), the grids of the sub-goal exit and are assigned value 0, grids of the obstructions are assigned 0 and others are assigned 1.
- (2) Set up a queue of grid $Queue<Grid>$, and add all the grid of sub-goal exit to the queue.
- (3) Get and remove the grid at head of the queue, and mark it as $Grid_{cur}$ (its $minDistance$ written as $minDistance_{cur}$). In the recurring pattern, check all the cellars surrounding the grid, and mark those surrounding cellars as $Grid_{adj}$ (its $minDistance$ written as $minDistance_{adj}$). If $available$ value of $Grid_{adj}$ is 1 and $minDistance_{adj} > minDistance_{cur} + 1$, then $minDistance_{adj} = minDistance_{cur} + 1$, and insert $Grid_{adj}$ to the tail of the queue. In the searching surrounding grids process, it needs to check the boundary which is ensure calculation include all the grids inside the subspace covered by distance map.
- (4) Repeat the step (3) until the grid queue is empty.

It can be seen from those steps, the generation of distance map is similar to Dijkstra algorithm, it takes those grids which are not occupied by the obstructions, and the node and the neighbor node are connected by edge which has weight 1. The generation process can be considered as the solution of shortest distance between the start node which is sub-

goal and the other nodes in the network. The distance map result is shown in Figure 2, and the Figure 3 shows that the distance map results with the obstruction.

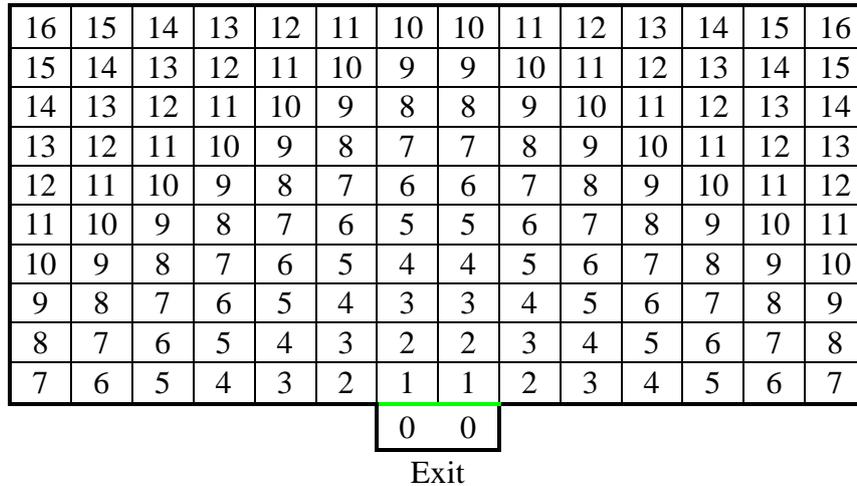


Figure 2. Example of Distance Map Result

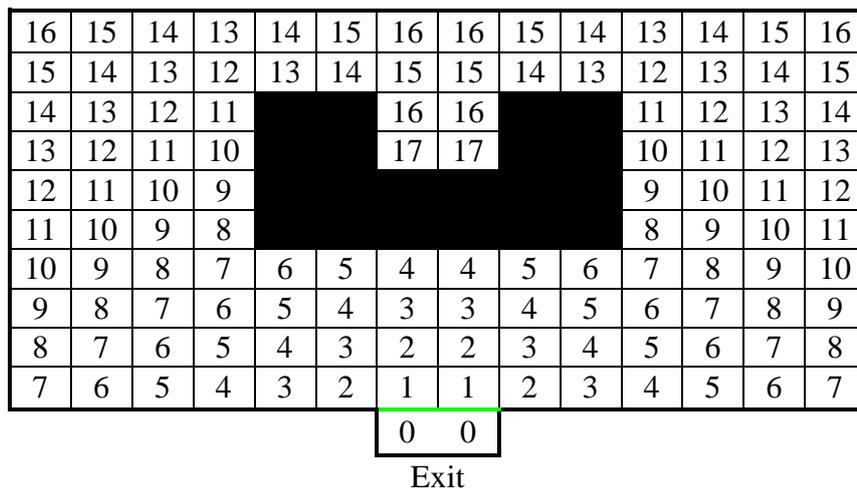


Figure 3. Example of Distance Map Results with the Obstruction

Compared with Figures 2 and 3, the distance between the grid behind the concave obstruction and sub-goal is longer than the other grids, it simulated the circumstance that pedestrian need to move afar to get round the obstruction.

3.4 Attraction Calculation

In this paper, we use the John von Neumann neighborhood [18] to demonstrate the experiment, and the neighborhood of each cellar is defined as the scope including the cellars in the three steps' range, the neighborhood is shown in Figure 4.

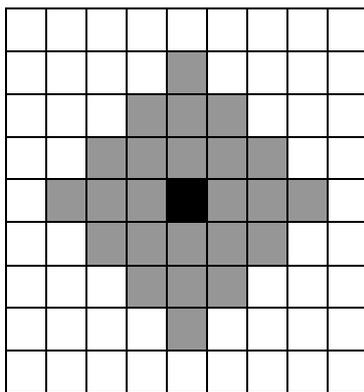


Figure 4. John von Neumann Neighborhood with Three Steps

In the John von Neumann neighborhood, the movement direction has five types: up, down, left, right and stay at current position. The cellular attraction model is introduced which is used to calculate the attraction and choose the target cellular when the shortest distances are the same. In the model people choose moving to cellular has high attraction which is different from the probability model. In one time step, five movements of cellular are corresponded to different attraction which is shown in Figure 5.

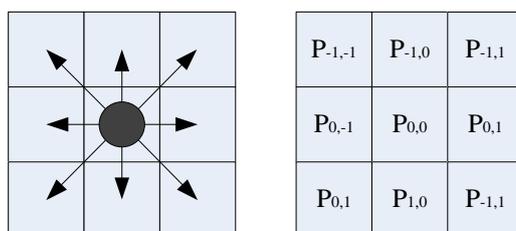


Figure 5. Moving Directions and Attraction of CA

Cellular attraction calculation condition: assume that *Grid1*, *Grid2* are the neighbor cellular of current people, and they has equal shortest distance to the sub-goal. If they are null, then calculate their attraction respectively.

Cellular attraction calculation method: assume that *Gridadj* is the cellular that needs calculate the attraction and assign the initial value 0. Check the neighbor of *Gridadj*'s according to priority, if any neighbor is null and its shortest distance less than *Gridadj*'s, then $Attraction=Attraction+5$; if any neighbor is null and its shortest distance equal to *Gridadj*'s, then $Attraction=Attraction+1$; if any neighbor is null and its shortest distance larger than *Gridadj*'s, then Attraction doesn't change; if any neighbor is not null, then Attraction doesn't change.

In the calculation process, $Attraction+5$ is to ensure people choose the cellular has small shortest distance. $Attraction+1$ to ensure people choose the cellular has low crowd density.

Cellular status update rules include:

- (a) If neighbor has only one *Gridadj*, and it is up to requirements: the shortest distance ($minDistanceadj$) less than the shortest distance of current cellular, and it is null ($available=1$), then choose the *Gridadj* as the target cellular; if there are multiple cellars meet the requirements, then calculate their attraction and choose the cellular that has max attraction as target cellular; if there are multiple cellars meet the requirements and they have same attraction, then choose the random one.
- (b) If there are no cellars meet the requirements in (a) but there are cellars meet following qualifications: the shortest distance ($minDistanceadj$) to the sub-goal equal to the shortest distance ($minDistancecur$) of current cellular, and current cellular is null

(*available=1*), then calculate the attractions of cellars, and choose the cellar that has max attraction as target cellar; If there are multiple cellars meet the requirements and they have same attraction, then choose the current cellar as target cellar. It is considered that people choose not to move to conserve energy when the attractions are the same.

- (c) If there are no cellars meet the requirements in (a), (b) and there do not exist the cellar that has null value neighbor (*available=1*) and the shortest distance less than the current cellar's shortest distance minus 1 ($minDistance_{cur}-1$), but there are cellars meet following qualifications: the shortest distance ($minDistance_{adj}$) to the sub-goal greater than the shortest distance ($minDistance_{cur}$) of current cellar which has null value, and there are neighbor's shortest distance is less than current cellar's shortest distance in the two steps of John von Neumann neighborhood. People choose the cellar that meets the requirements as target cellar. If there are multiple cellars meet the requirements, and then choose a random one as target cellar. It is simulated that people prevent from the jam and make a step backward.
- (d) It will be conflicts when more than one people have same target cellars, the solution is equal probability choose one people to move and others stay in position.

4. Experiment and Analysis

4.1 Experiment Initial Parameters Description

The experiment environment is shown in Figure 6a. The floor plan in this case has three subspaces and three sub-goals, and there are four rectangle obstructions marked in gray. The initial position of 50 pedestrians is randomly distributed in the floor plan, which is shown in Figure 8b[11]. Base on the sub-goal CA model we discussed, we perform a multiple subspace evacuation simulation which means that pedestrian need to pass through many subspaces before they get to the final exit. In this case, pedestrian in subspace1 will set the sub-goal to *sub-goal3* after they get to the *sub-goal1*, and pedestrian in *subspace2* will set the sub-goal to *sub-goal3* after they get to the *sub-goal2*.

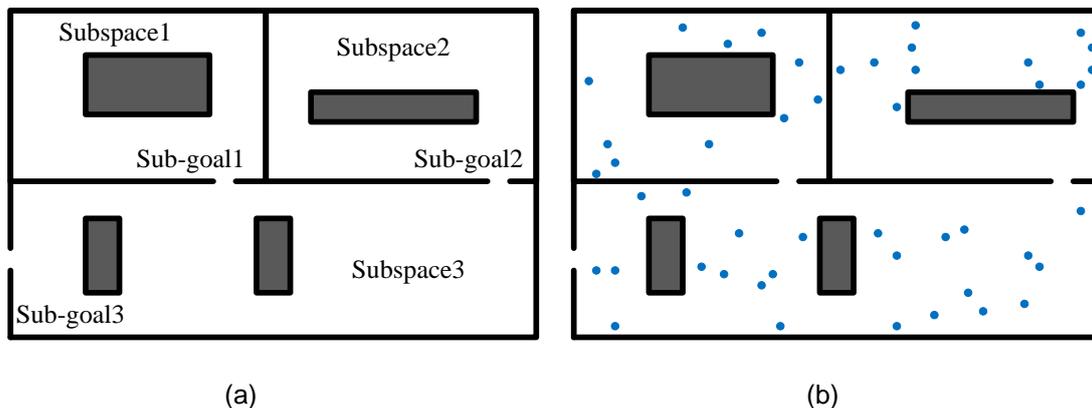


Figure 6. Floor Plan and Initial Position of Pedestrians

4.2. Result of Evacuation Process

According to the experiment environment in 4.1, we simulate the evacuation process by sub-goal CA model and traditional CA model respectively. The evacuation process during the 14th time step is shown in Figure 7. Result in Figure 7a is based on sub-goal CA model, and pedestrians in subspace1 and *subspace2* are moving towards *sub-goal1* and *sub-goal2* separately. This result conforms to our expectation. In Figure 7b, there are

several pedestrians circled by red ellipse. Because the target of those pedestrians is *sub-goal3* and they do not walk towards *sub-goal1* and *sub-goal2* in priority. This result shows that traditional CA model will take more time to finish the evacuation process.

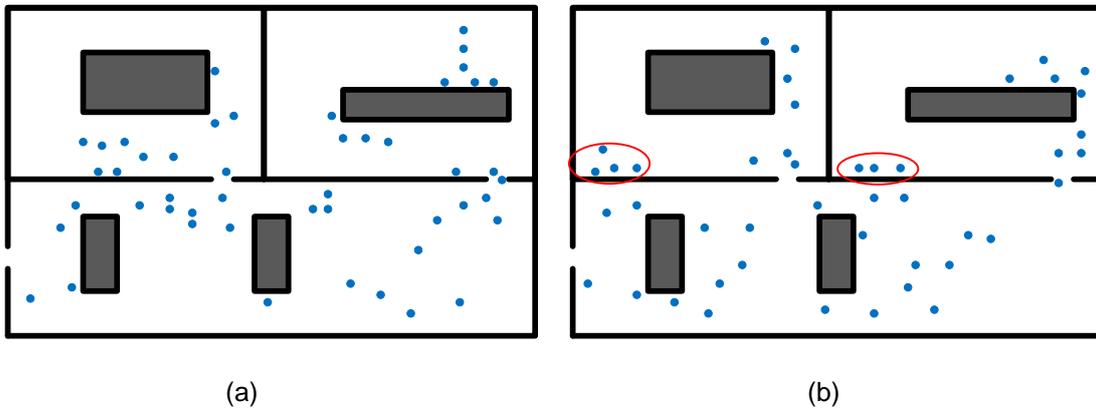


Figure 7. Evacuation Process during the 14th Time Step

4.3 Result of Evacuation Efficiency

In order to compare the evacuation efficiency more precisely, we calculate the evacuation time of sub-goal CA model and traditional CA model respectively. Based on the experiment environment in 4.1, we set the population of evacuation from 40 to 130 which divided into 10 groups, and we will run each group 10 times and take the mean to eradicate the discrepancies.

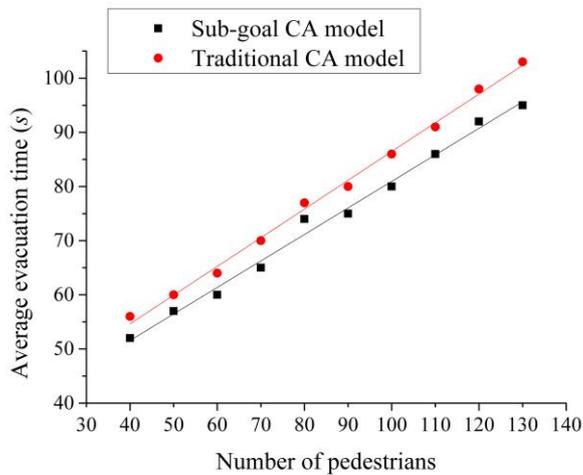


Figure 8. Average Evacuation Time with Subspace

Figure 8 shows two fitting straight lines of average evacuation time by sub-goal CA model and traditional CA model. The height of sub-goal CA model's fitting straight line is lower than traditional CA model's, which indicate that the average evacuation time of sub-goal CA model is shorter than traditional CA model.

To eliminate the difference between sub-goal CA model and traditional CA model, we set up a floor plan without subspace to redo the experiment. The floor plan only has one exit and shown in Figure 9a. We also use the same pedestrian group to calculate the average evacuation time, and the result is shown in Figure 9b.

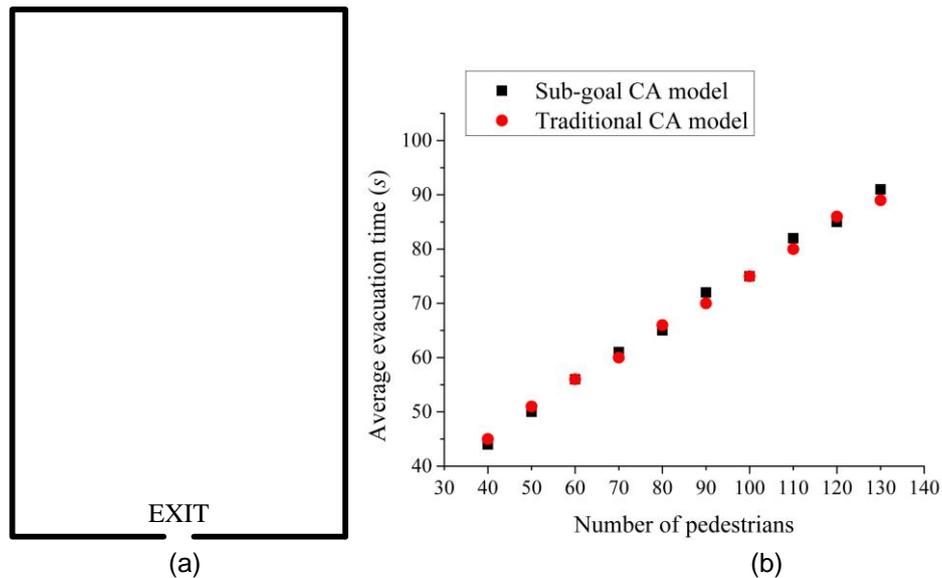


Figure 9. Average Evacuation Time without Subspace

From Figure 9b we can see that the evacuation efficiency is nearly the same between sub-goal CA model and traditional CA model when there is no subspace in the floor plan.

5. Conclusion and Future Works

This paper details an evacuation simulation method based on sub-goal CA, which uses CCRP algorithm to calculate the shortest path and distance map to indicate evacuation floor plan. In the evacuation process, evacuation spaces need to be reasonably demarcated and subspaces with the sub-goal are generated. When people moving in the subspace, they consider the shortest path to the current sub-goal instead the global path, and people pass around the obstruction by the distance map.

However, the building interior evacuation process in reality is very complex, and it is affected by many factors [1,2,3,4]. In the future, we will focus on the personality of individual pedestrian such as age and height in evacuation process by multi-agent model. On the other hand, we need to consider the dynamic environment of building interior, such as: fire spreading, smoke diffusion. In summary, the evacuation simulation should as approximate the reality as possible.

Acknowledgements

The authors hereby desire to express our indebtedness to anonymous reviewers for their valuable and constructive comments. This work is supported by the Zhejiang Provincial Natural Science Foundation of China (LQ13D010001, LY13F010006, LY16F010015), the Natural Science Foundation of China (61202171), the Program for Zhejiang Leading Team of Science and Technology Innovation (2011R50019), the Department of Education of Zhejiang Province's Research Project (Y201430369), and the Soft Science Research Project of Zhejiang Province (2014C35060).

References

- [1] J. Bohannon, "Building Safety: Directing the Herd: Crowds and the Science of Evacuation," *Science*, vol. 310, no. 5746, (2005), pp. 219–221.
- [2] A. L. Hammond, "Earthquakes: An Evacuation in China, a Warning in California," *Science*, vol. 192, no. 4239, (1976), pp. 538–539.
- [3] "Evacuation of the very Young," *Nature*, vol. 146, no. 3705, (1940), pp. 593–593, Nov. 1940.

- [4] E. Ronchi, "Disaster management: Design buildings for rapid evacuation," *Nature*, vol. 528, no. 7582, (2015), pp. 333–333.
- [5] H. W. Hamacher and S. A. Tjandra, "Mathematical Modelling of Evacuation Problems: A State of Art," *Fraunhofer-Institut für Techno-und Wirtschaftsmathematik, Fraunhofer (ITWM)*, vol. 24, (2001), pp. 1–45.
- [6] J. Von Neumann, *Theory of Self-Reproducing Automata*, 1st Ed. Champaign, IL: University of Illinois Press, (1966).
- [7] B. S. Kerner, S. L. Klenov, and D. E. Wolf, "Cellular automata approach to three-phase traffic theory," *Journal of Physics A: Mathematical and General*, vol. 35, no. 47, (2002), pp. 9971–10013.
- [8] D. E. Wolf, "Cellular automata for traffic simulations," *Physica A: Statistical Mechanics and its Applications*, vol. 263, no. 1–4, (1999), pp. 438–451.
- [9] B. S. Kerner and S. L. Klenov, "A microscopic model for phase transitions in traffic flow," *Journal of Physics A: Mathematical and General*, vol. 35, no. 3, (2002), pp. L31–L43.
- [10] Q. Lu, B. George, and S. Shekhar, "Capacity Constrained Routing Algorithms for Evacuation Planning: A Summary of Results," in *Advances in spatial and temporal databases*, no. 81655, (2005), pp. 291–307.
- [11] C. Burstedde, K. Klauck, A. Schadschneider, and J. Zittartz, "Simulation of pedestrian dynamics using a two-dimensional cellular automaton," *Physica A: Statistical Mechanics and its Applications*, vol. 295, no. 3–4, (2001), pp. 507–525.
- [12] A. Kirchner and A. Schadschneider, "Simulation of evacuation processes using a bionics-inspired cellular automaton model for pedestrian dynamics," *Physica A: Statistical Mechanics and its Applications*, vol. 312, no. 1–2, (2002), pp. 260–276.
- [13] A. Varas, M. D. Cornejo, D. Mainemer, B. Toledo, J. Rogan, V. Muñoz, and J. A. Valdivia, "Cellular automaton model for evacuation process with obstacles," *Physica A: Statistical Mechanics and its Applications*, vol. 382, no. 2, (2007), pp. 631–642.
- [14] R. Alizadeh, "A dynamic cellular automaton model for evacuation process with obstacles," *Safety Science*, vol. 49, no. 2, (2011), pp. 315–323.
- [15] K. Zhiqiang, Y. Chongchong, T. Li, and W. Jingyan, "Simulation of evacuation based on Multi-Agent and cellular automaton," in *2011 International Conference on Mechatronic Science, Electric Engineering and Computer (MEC)*, (2011), pp. 550–553.
- [16] D. X. Xuan, "Theoretical and Experimental Study of a Multi-grid Model for Pedestrian Evacuation," Doctor dissertation: University of Science and Technology of China, (2009).
- [17] J. D. Averill and W. Song, "Accounting for emergency response in building evacuation: modeling differential egress capacity solutions," Working Paper, Building and Fire Research Laboratory, National Institute of Standards and Technology, Technology Administration, US Department of Commerce, (2007).
- [18] J. von Neumann, "The general and logical theory of automata," *Cerebral mechanisms in behavior*, vol. 1, no. 41, (1951), pp. 1–2.

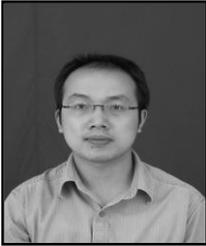
Authors



Zhiwen Hu. He received his B.S., M.S., and Ph.D. degrees in 1996, 2003, and 2006, respectively, from China University of Geosciences, Wuhan, China; Hefei University of Technology, Hefei, China; and Chinese Academy of Sciences, Hefei, China. From 1996 to 2006, he worked at the School of Computer Science and Technology, Huaibei Normal University, Huaibei, China, as a Lecturer and an Associate Professor. From January 2007 to August 2013, he worked at Wenzhou University, Wenzhou, China. From Feb. 2015 to Feb. 2016, he worked at University of Victoria, Canada, as a visiting research fellow. He is currently an Associate Professor at the Zhejiang University of Media and Communications, Hangzhou, China. He is coauthor of more than 80 articles, which mostly were published in high-profile journals.



Li Zhu. She received his B.S., M.S., and Ph.D. degrees in 2004, 2006, and 2010, respectively, from Wuhan University, Wuhan, China; she has been working continuously in WenZhou University as a lecturer until now. She is coauthor of more than 40 articles, which mostly were published in high-profile journals.



Huaqing Mao. He holds his PhD from Wuhan University, China, in 2010. He then joined the Information Security Laboratory at the Wenzhou University, China, and his main research interest is the tall building interior emergency response and evacuation. More recently, he has been focusing on the model construction and optimization of tall building.