MATH 435 Project 2
Louvre Evacuation Plan

Fu, Yiqi; Sun, Yimeng; Li, Jingya; McAliney, Seumus

April 11, 2019

Contents

1 Introduction

1.1 Background Material

The *Louvre Museum* is one of the world's largest art museums and is also a historic monument in Paris, France. The floor space is approximately 72,735 square metres. The most recent terrorist attack at the *Louvre Museum* was the *2017 Paris machete attack* [3], which happened near the public entrance to the Museum. This attack caused confusion and exacerbated the decline in tourism. The *Louvre Museum* is a crowded place with really high population density. Based on the online application *Affluences* [5], the daily number of visitors to the Louvre is around 5000 people. Our group was interested in finding an effective way to evacuate visitors and working staff from the Museum safely and quickly [6].

The main entrance of the *Louvre Museum* is on the -2 Floor, which is also known as the *Pyramid Main Entrance*. Our team separated the whole model into two parts. The first part was for each floor, we used MATLAB to transform the floor into a grid and calculate the distance between each grid point and the exits, which allowed visitors to find the nearest stairs (there are no elevators available in an emergency). This created an effective path for each guest to get to the -2 Floor. The second part was to account for when they got to exceptionally high density areas (bottlenecks), since there are often many people in one section of the *Louvre* that may limit movement towards the exits. For example, when visitors all get to the -2 Floor, people will have to wait in a queue, and we calculated the time it took for them to pass this bottleneck.

2 Methods

2.1 Introduction to Cellular Automaton

Cellular automaton is the main model we used in this project. The rule for a 1-dimensional cellular automaton is that there are two possible states (0 and 1). The state of a cell x_i^t in the next generation depends only on the current state of the cell and its two immediate neighbors x_i^{t-1}, x_{i+1}^{t-1} and x_i^{t-1} , where t is the time step, and i is the index in each generation. This 1-dimensional rule is the core principle for the general cellular automaton method in this project.

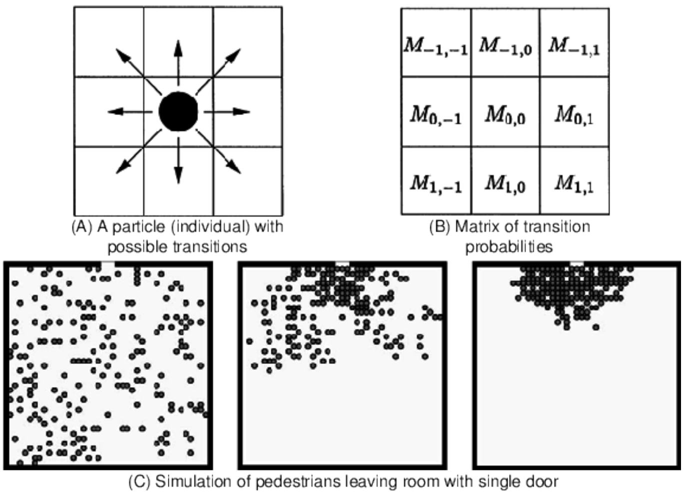


Figure 1: General view of the Cellular automation [8]

Basically, a mover needs to choose one of the eight neighbors as the optimal position of the next step, which depends on the "motivation" F. The motivation F is a function which determines the next cell in which the person should appear in the next time step. In our basic model we used the distance between the cell and the exit as the only motivation. The longer distance represents the more dangerous position, and people always head to the less dangerous position corresponding to the smallest distance. If the coordinate of a person is (x, y), the coordinate of the exit is (a, b), so the distance between the person and the exit is [4]:

$$F = \sqrt{(x-a)^2 + (y-b)^2} \tag{1}$$

The algorithm serves the following purposes:

- Judge the availability of his eight neighbors
- Calculate the value F of his available neighbors
- Sort the value F and choose the cell whose value is smallest as the destination of the person.

In this project, we applied this model to the *Louvre* in order to introduce a simulation which describes how the population is distributed and moving over time. We used MATLAB to transform each floor of the Louvre into a grid of squares [2].

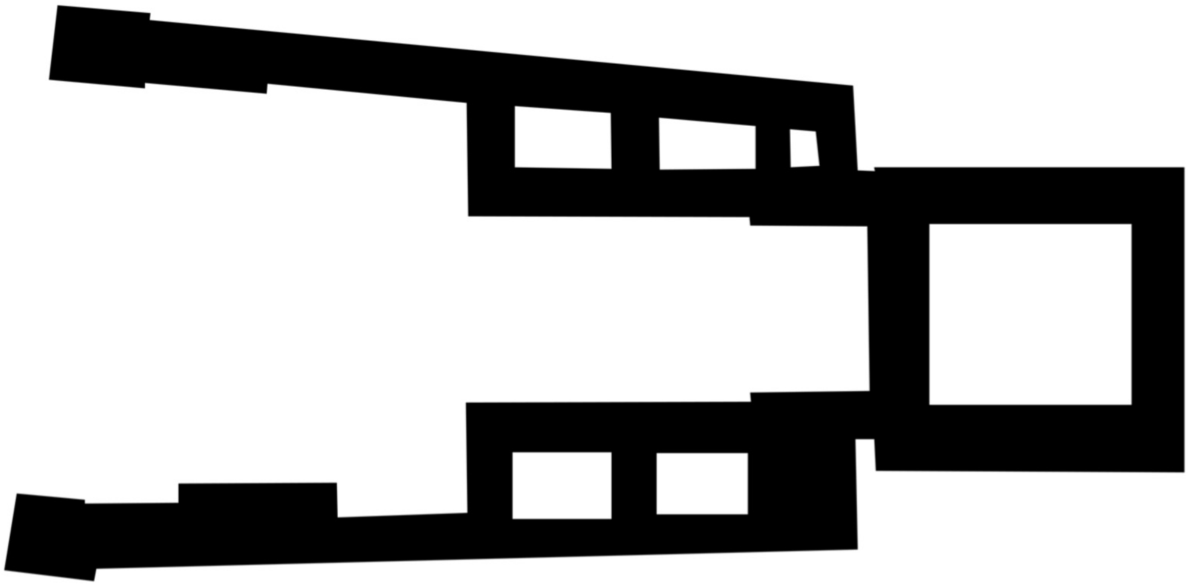


Figure 2: Binary map of Louvre

For each square in the map, we calculated the danger level for the eight squares around it. The eight neighbours also stands for the eight directions in which the people in the center square can travel. Comparing the danger levels, we found the direction with the lowest danger level, and then in the next generation, the people in the center moved in that direction. In this way, we found the safest way for a visitor to escape.

2.1.1 Applying Our Model to the Louvre

Within the model we developed, we made a few assumptions to make the simulation possible. We assigned a higher danger level to individual squares in our simulation based on distance from exits, and whether or not they were part of a potential bottleneck. In a real world scenario, however, it might be necessary to include higher danger levels to account for obstructions that may be present in the building during an evacuation, such as fires, or assailants with weapons. The equation for the danger level is:

$$D = k \times F \tag{2}$$

where k is the constant and F is the distance from a certain square to the exit, which could be obtained from equation(1). In this project, we chose $k = 1$ since there is no emergency inside of the *Louvre*. If there is some accident happening in some grids, higher k will be applied around those grids.

2.2 Variables and Assumptions

Based on the online application *Affluences* [5] and a map of the *Louvre Museum*, we made a few guesses about the population. The application gave us the data about the population flow in a real scenario, however, we could only obtain the waiting time from this app, so this data only helped a little in our project. Here are the assumptions:

- The whole population density in the *Louvre Museum* at a certain time period is 0.1, which means for each 10 squares there is one person, and the whole population is randomly produced each time.
- The bottlenecks stand for the queuing areas that have higher danger levels than the other normal grids.
- Since the population is randomly generated under a specific density, the bottlenecks are also randomly produced each time.
- Visitors on higher floors will need to join the evacuations on lower floors. (i.e. Visitors on the 5th floor will first get to the 4th floor, and use the stairs together to get to the 3rd floor. The evacuations are accumulated through each floor.)

2.3 Coding

Some of the code about cellular automaton was referenced from Git-Hub [7].

2.3.1 Rules for Drawing the Map

We made a data frame for the map, the index of columns stands for the x-axis, the index of rows stands for the y-axis. The value in the frame stands for the status of that certain square. In total there are four different statuses [1].

- 0: There is an impassable barrier.
- 1: There is a person in the grid.
- 2: Normal area where visitors may be randomly distributed.
- 3+: The exits.

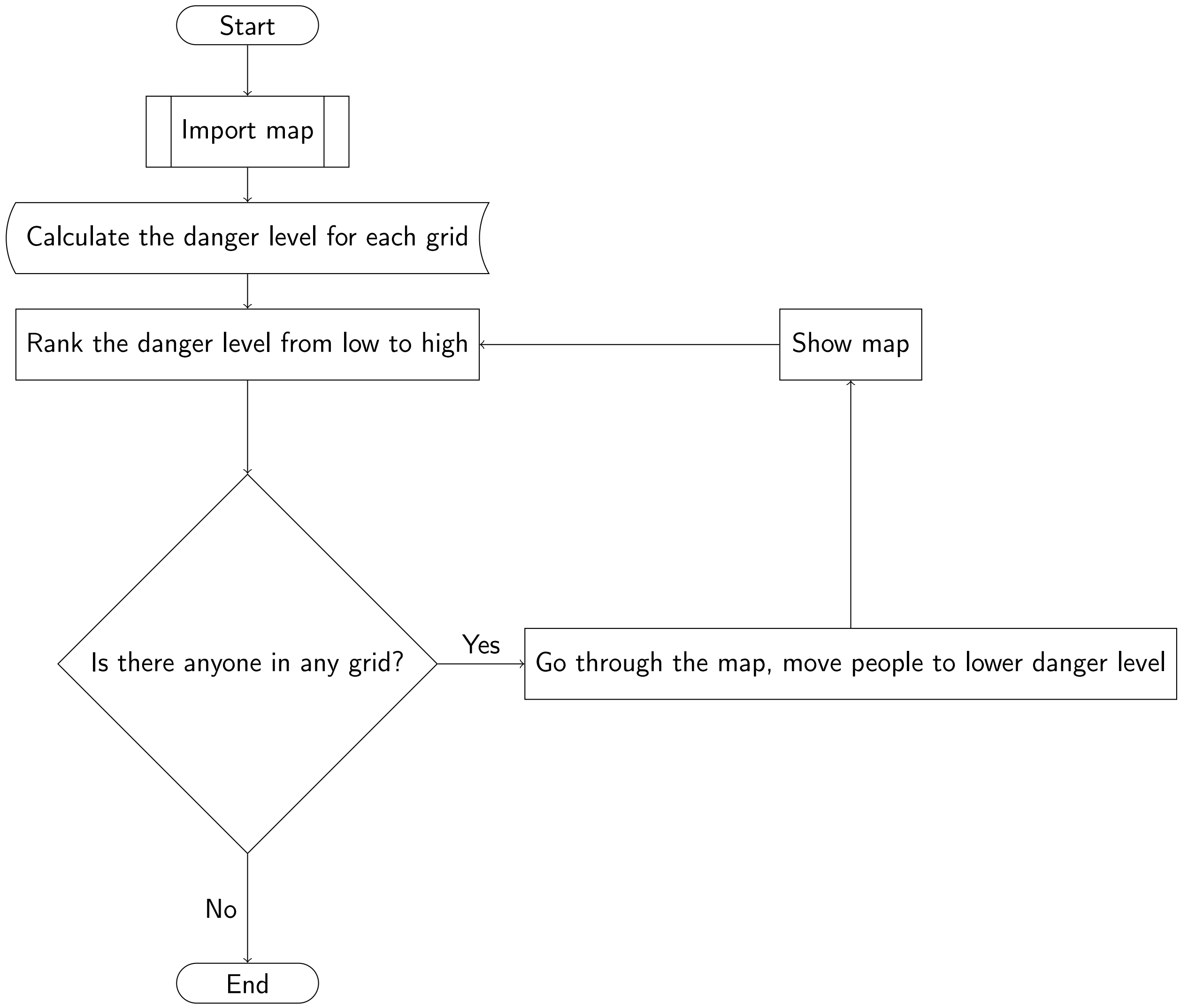
2.3.2 Danger Level

In order to find the shortest way to escape from the *Louvre*, we calculated the danger level for each grid. The danger level here is the shortest distance from each grid to the nearest exit.

2.3.3 Rules

- Walk randomly in the direction where the risk is strictly lower, if this is not an option, walk randomly in the direction where the risk is equal.
- Go through the whole map, following the danger level to the smaller values, finding the way to the smallest danger level grid square, and then moving o that square.
- Traveller can choose four moving directions (up, down, right, left).
- People disappear when moving to an exit (exit safely).

2.4 Flow Chart



3 Results

Based on the cellular automaton we are able to present a simple model to simulate how to evacuate the whole population from the *Louvre Museum*.

3.1 Evacuation Simulation

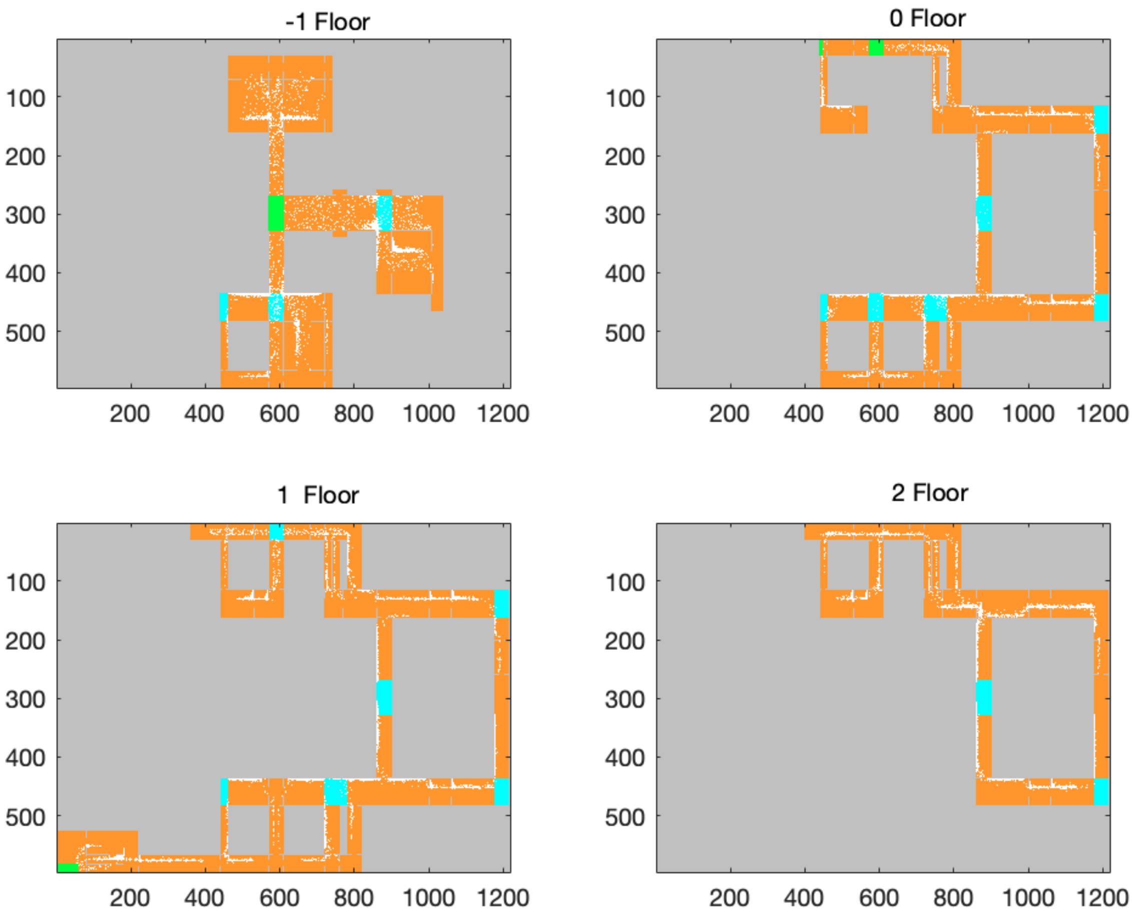
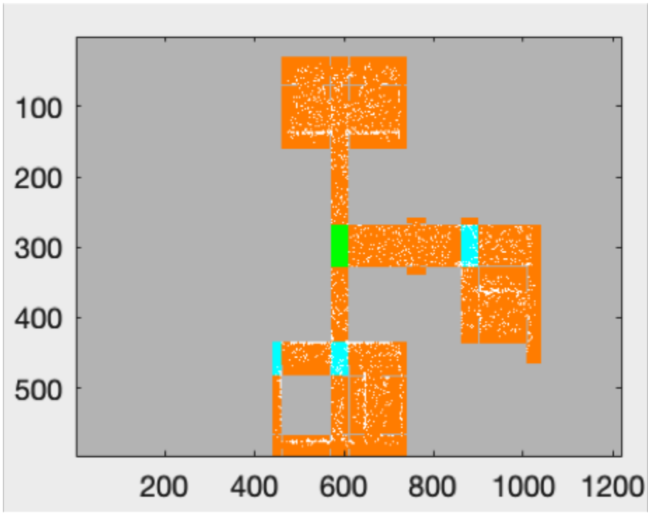
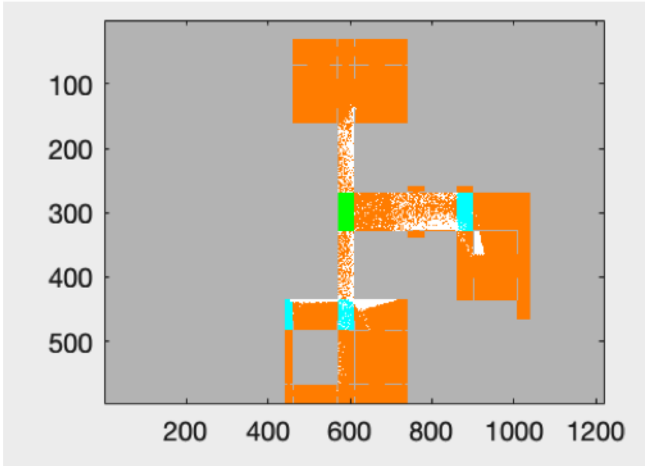


Figure 3: Evacuation Simulation for Each Floor

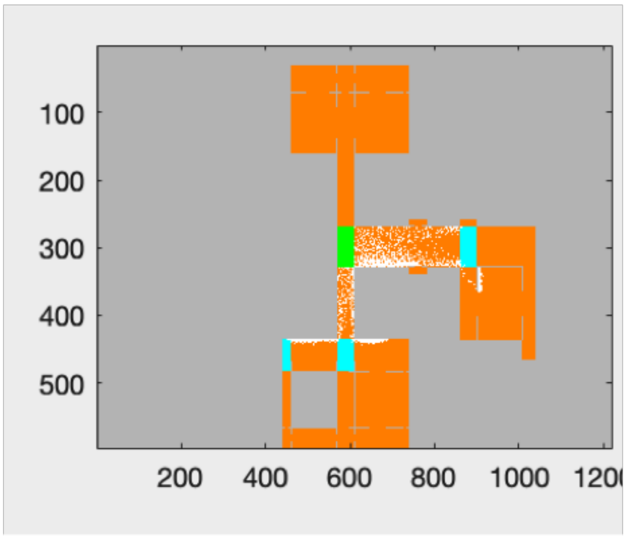
In Figure 3, we successfully simulated a dynamic progress for travellers escaping from the Louvre [3].The orange areas are normal places, the light blue squares are the exit stairs. The green squares in the -1 Floor represent the general exit, where visitors leave the Louvre.



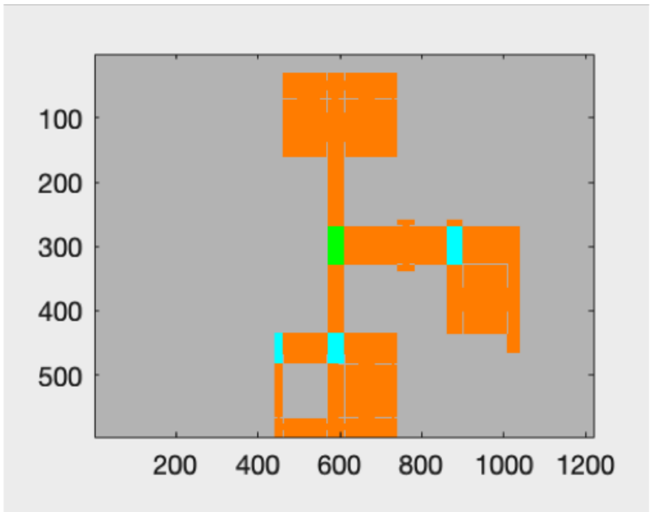
(a) T = 1s



(b) T = 500s



(c) T = 1000s



(d) T = 1812s

Figure 4: The Dynamic Evacuation Process in -1 Floor

These four subplots in Figure[4] give a dynamic view in terms of the population change in -1 floor. Since we cannot use video to show the progress in the report, the subplots may help to understand the progress.

3.2 Comparison

In order to have a visible view of how the time decreases through applying the cellular automaton algorithm, we compared the time cost before the algorithm and after the algorithm.

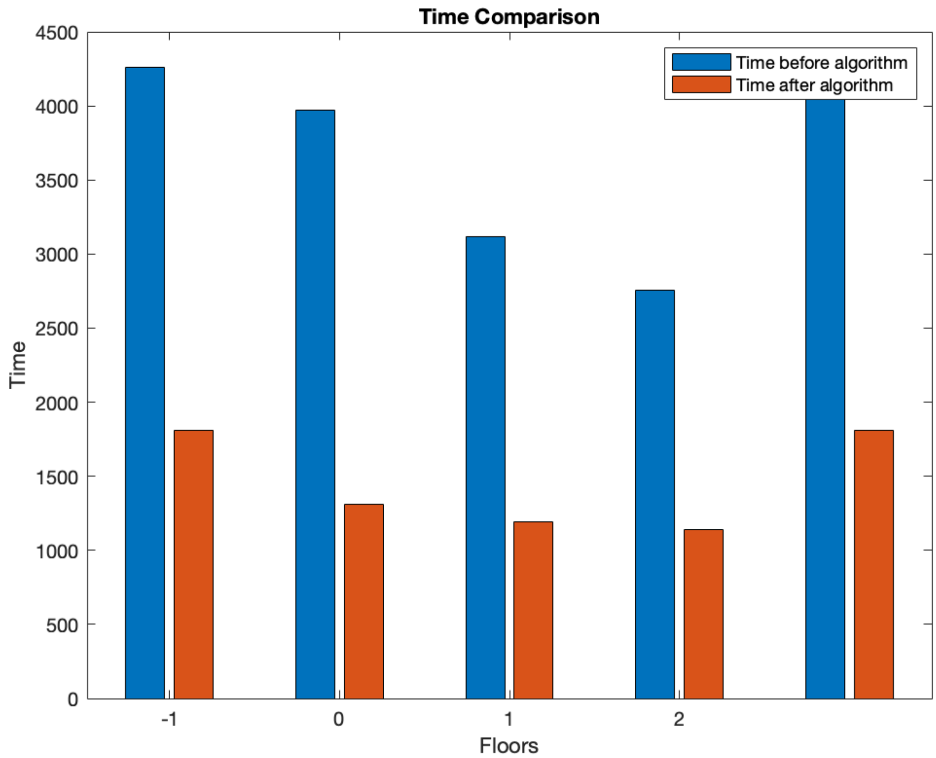


Figure 5: Evacuation Time Comparison

Here we used a shortcut to calculate the time before the algorithm. The time was calculated by: dividing the distance between each person and a random exit by the same velocity in the algorithm. The distance is just the shortest straight-line distance, where people may be able to go through a wall, which is definitely unpractical. For real life, the time will be longer for visitors to escape before using the cellular automaton algorithm. The time here stands for total time for the population evacuate from each floor. In the rightmost column in Figure 5, clearly, there is a huge decrease in the total time cost, which decreased from about 4300 seconds to 1800 seconds. Thus, the cellular automaton algorithm effectively find a shorter way for travellers escape from the *Louvre*.

Here is another plot shows the population changes according to time:

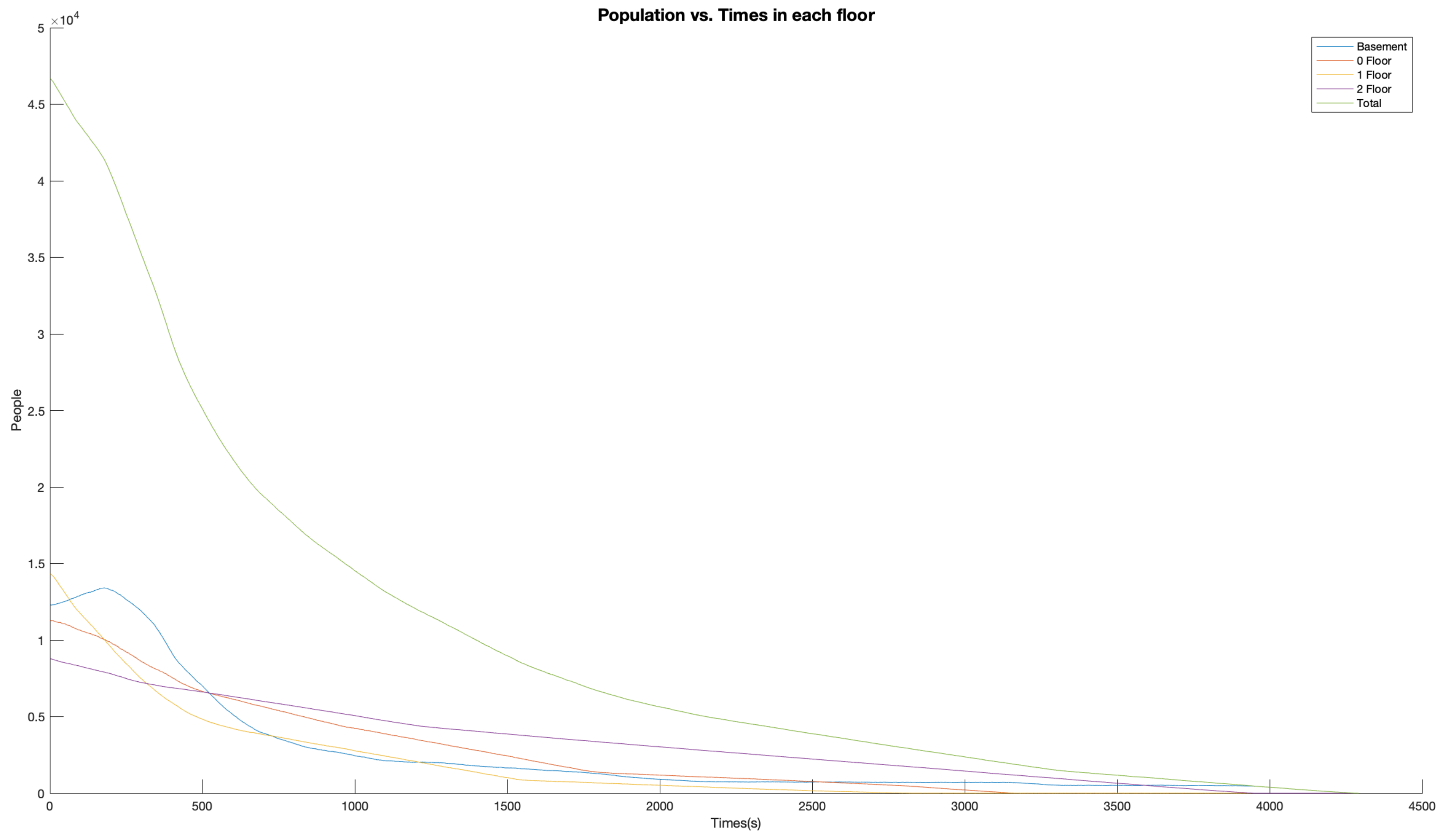


Figure 6: Population Change in Each Floor

This figure 6 shows how the population distributed in each floor before the algorithm.

After several iterations, we found that the population was always crowded in room 823, 803, and 909. Therefore, by monitoring these three rooms we got another plot[8], which could help the museum staff to evacuate the crowd. For example, at around 400s, the population in room 803 reaches its peak. By monitoring the population flow, the staff could guide the crowd to another exit, which means a huge deduction in the queuing time.

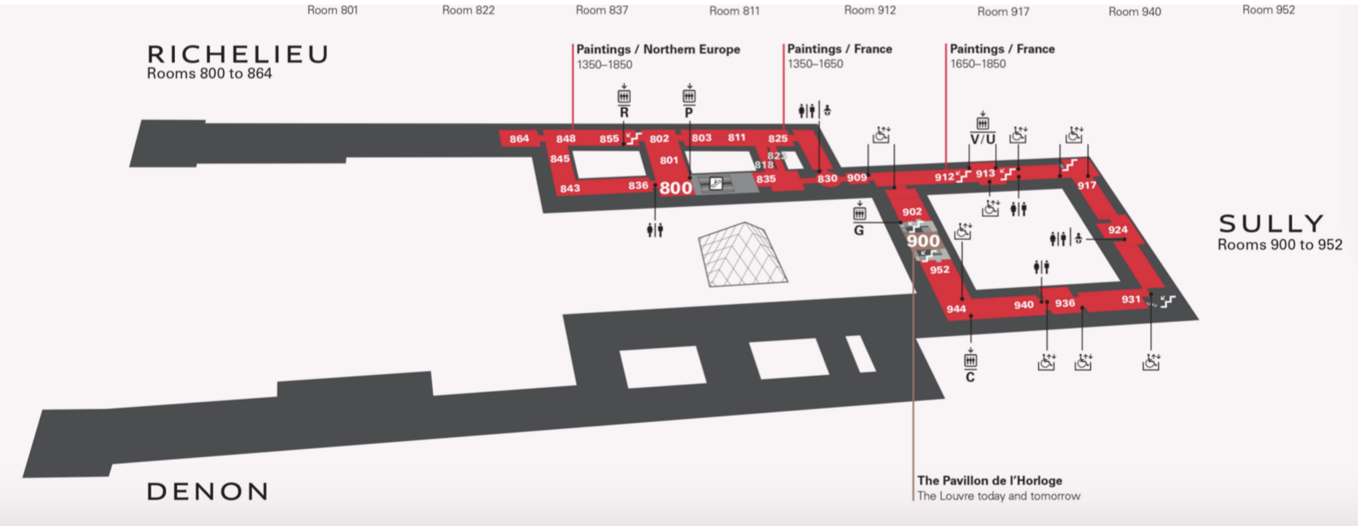


Figure 7: 2ND Floor Plan

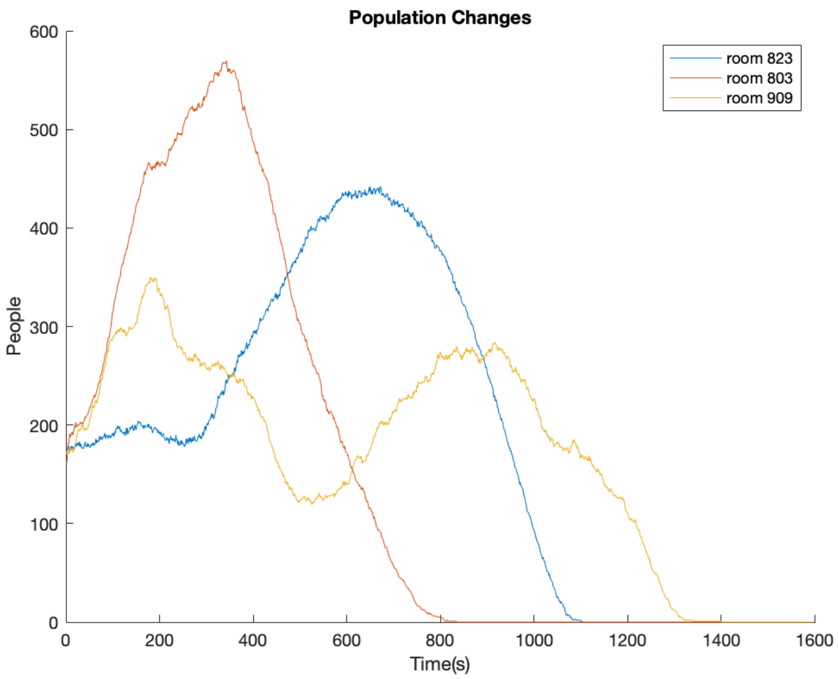


Figure 8: Population Change in Some Rooms

4 Conclusion

In this project we were tasked with creating an effective evacuation plan for the Louvre, one of the world's largest and most visited museums. We utilized Cellular Automaton to create an algorithm to direct each individual's movement based on their location in the museum and the density of people in their immediate vicinity. We developed our model in MATLAB. This model provided an efficient evacuation plan for the Louvre, and when compared to current evacuation plan exit times, our model proved to be much quicker. Within our model we identified 50 different areas with higher danger levels than the other areas of the museum. These were the bottlenecks, or areas where their would undoubtedly be a high density of people at any given time. However, this was the only criteria for assigning a higher danger level to a square. In a real world scenario, there could be any number of reasons for certain areas of the museum to be more dangerous than others. This will need to be addressed in future work on the project.

4.1 Application of the Model

Part of the problem posed to us in the MCM contest was to implement the model we came up with into the Louvre's evacuation procedure. While we did not have much time to address this aspect of the problem, we briefly discussed some solutions. It would be in the best interest of the museum to have this model running in real time, with live updates of each individual's location within the museum. Should an emergency occur, security personnel will be able to view the map and direct everyone to the proper exits. We tentatively recommend an application, such as *Affluences* [5], which will provide a map for each individual on their smart-phones, which they can then follow to safety. However, this procedure gives rise to many flaws and should be extensively tested before implementation.

5 Future Work

5.1 Modify Model for Real World Scenario

In our method, we only used 50 areas with higher risk and analyzed the time needed for the people in these areas to evacuate. We think our method needs some improvements so that it can be applicable for a real-world scenario which has a lot of variables that would influence the effectiveness of our evacuation plan. However, we cannot form a more comprehensive plan due to the limitation of knowledge of the Louvre and inability to run our model into the experiment. In the long run, We suggest that we can take all areas of the Louvre into account and analyze it with our model instead of only 50 areas. In this way, we could cover everyone in this museum and form a more effective and comprehensive evacuation plan. Also, we hope we could test the actual performance of our model in the Louvre or some experimental sites that are nearly the same as the Louvre if possible. With more experiment of our model, we will be able to find some variables that can be adjusted for different destruction caused by the terrorist attack and then we can label the variables for the museum evacuation team. With the refined model, the museum evacuation team will respond more quickly for any kinds of terrorist attacks that happen in the museum.

5.1.1 Adaptation to Different Situations

In this project, we apply the cellular automaton model to create an evacuation plan with an assumption of the terrorist attack happen outside of the Louvre. We propose that the danger level, the kind of status, and the status in the data frame can be changed, to make the model applicable for different destruction caused by the terrorist attack. For example, we can reduce the number of statuses which represent entrances in the museum and make these entrances only available for the entry of police force or medical support. Also, we can change the constant of our danger level equation for some specific square in the museum because these places are close to the terrorist or in their sight. In addition, in order to ensure the evacuation plan formed by our model can be used in practice, we suggest that the security personnel in the Louvre should equip a secured phone with an application which will provide in-time evacuation plan so that they can guide the visitors to evacuate. The evacuation plan will form by our model and adjust by the emergency response team of the Louvre.

6 Distribution of Work

Yiqi Fu:

- Future Work
- Brainstorm
- PowerPoint Creation

Jingya Li:

- Brainstorm
- Method Part
- Result Part
- Mathlab Coding

Yimeng Sun:

- Brainstorm
- Background
- Introduction
- Introduction in Method Part

Mcaliney Seumus:

- Conclusion
- Brainstorm
- PowerPoint Creation

References

[1] City planning based on cellular automata. https://blog.csdn.net/sk18192449347/article/details/77312160.

[2] Interactive floor plans. https://www.louvre.fr/en/plan, Jun 2016.

[3] 8.1 million visitors to the louvre in 2017. http://presse.louvre.fr/8-1-million-visitors-to-the-louvre-in-2017/, Jul 2018.

[4] Cellular automaton. https://en.wikipedia.org/wiki/Cellular_automaton, Mar 2019.

[5] AFFLUENCES. Affluences. https://www.affluences.com/louvre.php.

[6] REPORTERS, T. Terror attacks in france: From toulouse to the louvre. https://www.telegraph.co.uk/news/0/terror-attacks-france-toulouse-louvre/, Feb 2017.

[7] SIMONDLEVY. simondlevy/ca. https://github.com/simondlevy/CA, May 2016.

[8] ZHOU, S., CHEN, D., CAI, W., LUO, L., LOW, M., TIAN, F., TAY, V., WEE SZE ONG, D., AND D. HAMILTON, B. Crowd modeling and simulation technologies. *ACM Transactions on Modeling and Computer Simulation 20* (10 2010), 20.