Simulation of Emergency Evacuation
Based on Agent-based Model

Team-06: Zihong Hao, Qilin Li, Chen Lin, Hangzhang Liu, Mian Wu[1]

[1]zhao48, qli460, clin449, hliu716, mwu385@gatech.edu

GitHub Repo: https://github.gatech.edu/zhao48/CSE6730-Project

Abstract

In the face of unpredictable situations such as fires, disasters, and toxic gas leaks, emergency evacuation in public areas has received attention from society. This study represents a mathematical model based on pedestrians' movement factors for emergency evacuation. In this study, we employ the pedestrian evacuation model built on the social force model with modified settings on driving forces and interaction forces between human beings and between individuals and obstacles. The goal of the study is to adjust public space designs to reduce evacuation time. We plan to demonstrate the effectiveness of the model through agent-based simulation by changing the forces between agents and the simulation of the evacuation process under real-world evacuation network settings.

1 Description of the system being studied

Crowed and group simulation are essential topics in emergency management. In the real world, the situation becomes much more serious because pedestrian behaviors always involve life safety and social orders(schools, airports, etc.) which exposes real danger to people. In this project, we construct a simulation model which reflects people's behavior in an emergency evacuation. Our model is based on people's interactions with others, obstacles, and exit choices. In our project, we will use the social force model [HM95]. In this model, when an organism receives a stimulus through its senses, it responds by selecting behavior that aligns with its personal goals and preferences, based on a range of possible behaviors. The ultimate goal of this behavior selection is to maximize utility or benefits to the organism. Due to pedestrians' familiarity with their typical situations, their responses are often instinctive and based on their past experiences of which reaction will be the most effective.

In detail, The individuals in the positions follow the rule through the social force model in moving to the location provided by the navigation agent. The entire process is repeated until the exit is reached. We have considered many factors in this experiment and designed a working frame as follows:

(1) People move or try to move considerably faster than normal. (2) Individuals start pushing, and interactions among people become physical in nature. (3) At exits, arching and clogging are observed in a semicircle, leading to jams. (4) People tend towards mass behavior, that is, to do what other people do. (5) Alternative exits can be chosen by distance and density of crowed.

2 Literature review

In emergency preparedness, the study of how people can quickly and securely leave a structure or an area during an emergency is known as evacuation modeling. Evacuation modeling uses mathematical models to analyze and optimize emergency evacuation processes. There are various types of models have been developed, including agent-based models [CLQ96], Cellular Automata (CA) models [PM08], and fluid dynamic models [BG12].

The agent-based strategy, which treats each person as an agent and simulates their behavior during an evacuation, is a well-liked technique. Roan and Haklay conducted one of the initial studies [RHE11] that employed the agent-based approach for evacuation simulation, suggesting a model representing how a building would be evacuated in the event of a fire. The model considered elements, including the structure's architecture, inhabitant behavior, and impediments. The agent-based technique is an effective way of simulating evacuation that takes into account the actions of individual agents. This research shows how the agent-based technique may be used to simulate the evacuation process and assess the efficacy of various evacuation tactics. To increase the agent-based models' precision in foretelling occupant behavior in an emergency, more study in this area is required.

Another widely-used model, Cellular Automata (CA), can be applied to simulate how people move across a physical location over time to depict evacuations. A grid of cells is used to represent the actual space in this context, and each cell might be in one of the multiple states (e.g., occupied by an individual, empty, obstructed by an obstacle, etc.). One example is the work of Bandini [BCGV14] in 2014, who created a CA-based model for pedestrian dynamics, which serves as one illustration. They created rules controlling people's movement based on their environment and utilized a two-dimensional grid to simulate a pedestrian area. A variety of pedestrian behaviors, such as congestion, lane creation, and evacuation, may be accurately simulated by the model.

The movement of people may be modeled as a fluid flow in an evacuation scenario, with people acting as particles in the flow. The model examines parameters such as the density of people in various regions, the pace of movement, the accessible departure routes, and the features of the building or surroundings. Therefore, fluid dynamic models could provide an analysis of the efficiency of various evacuation plans and point up any possible dangers or bottlenecks. These models have been used to study various aspects of evacuation, such as exit capacity, congestion, and the impact of social norms and communication.

3 The conceptional model of based pedestrian evacuation

3.1 Our modified social forces settings

The social forces model was first established by Helbing [HM95]. the social force model (SFM) based on Newtonian mechanics from characteristics of collective behaviors. The social force refers to the force that one individual obtains from the environment (including humans and objects), whereas the physical force is the force that is directly applied to the individual.

Based on the different motivations of pedestrians and impacts from the environment, SFM has four forces: (1) the driving force, (2) the interaction force between human beings, and (3) the interaction force between individuals and obstacles.

Our modified model considered the first three forces. The resultant force of the three forces impacts pedestrians and contributes to acceleration. The internal driving force guides the individual to move toward the target. However, before body contact, the exclusive force prevents individuals from colliding with one another. Specifically, to prevent individuals from colliding with obstacles we considered the smart strategy: if the agent meets the lines, the driving force will change to the best verticle of obstacles and then recover to initial settings (referring Section 3.3 for details). We define the resultant force of three forces as follows:

$$\overrightarrow{\xi}(t) = m_i \frac{d\overrightarrow{v_i}(t)}{dt} = \overrightarrow{f_i^0} + \sum_{j \neq i} \overrightarrow{f_{ij}} + \sum_{w} \overrightarrow{f_{iw}} \tag{1}$$

where m_i is the mass of pedestrian i, and $\overrightarrow{v_i}(t)$ is the actual walking velocity. Eq.(1) shows that the motion of pedestrian i is affected by four types of forces, which include the pedestrian's driving force $\overrightarrow{f_i^0}$, the interaction force between pedestrian i and the other pedestrians $\sum_{j(\neq i)} \overrightarrow{f_{ij}}$, the interaction force between pedestrian i and obstacles $\sum_w \overrightarrow{f_{iw}}$ when meeting the obstacle the driving force become 0 and modified interaction between pedestrian and obstacles appears. The position of pedestrian i changes under the interactions of the three forces.

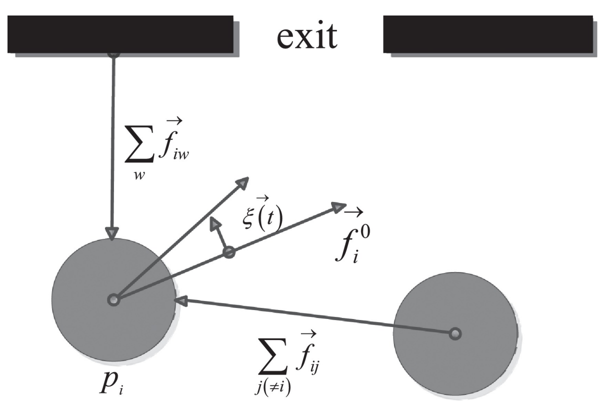

Figure 1: The enclosed area is the exit effect area. The blue bar on the black boundary indicates the exit while the blue arrow represents the radius of the affected area.[LLZ+18]

The pedestrian's driving force $\overrightarrow{f_i^0}$ is defined as

$$\overrightarrow{f_i^0} = m_i \frac{v_i^0(t)\overrightarrow{e_i^0}(t) - \overrightarrow{v_i}(t)}{\tau_i} \tag{2}$$

where m_i is the mass, and $\overrightarrow{v_i}$ is the actual velocity of pedestrian i. Moreover, the velocities were controlled by the velocities and the acceleration equations can be expressed as follows: $\overrightarrow{a_{il}^0} = \frac{v_{il}^0(t)\overrightarrow{e_{il}^0} - \overrightarrow{v_i}(t)}{\tau_i}$ and $\overrightarrow{v_i}(t) = \frac{d\overrightarrow{r_i}}{dt}$ where r_i represents current position.

Moreover, to model the interaction between pedestrians, we need to calculate repulsive force, body force, and the sliding friction force related to tangential motion. The psychological tendency of two pedestrians i and j must stay away to avoid overlapping. Here we model the repulsive interaction force that pedestrians stay away from each other by $A_i \exp\left[\left(r_{ij} - d_{ij}\right)/B_i\right]\overrightarrow{n_{ij}}$, where A_i and B_i are constants. $d_{ij} = \|\overrightarrow{r_i} - \overrightarrow{r_j}\|$ denotes the distance between the centres of mass. $\overrightarrow{n_{ij}} = \left(n_{ij}^1, n_{ij}^2\right) = \left(\overrightarrow{r_i} - \overrightarrow{r_j}\right)/d_{ij}$ is the normalized vectors between them.

Body force which counteracts body compression and the force which impedes relative tangential motion(sliding friction force) is essential for understanding the particular effects in panicking crowds. Body force can be model as $k\left(r_{ij} - d_{ij}\right)\overrightarrow{n_{ij}}$. The sliding friction force can be expressed as $\kappa\left(r_{ij} - d_{ij}\right)\Delta v_{ji}^t \overrightarrow{t_{ij}}$, where $\Delta v_{ji}^t = \left(\overrightarrow{v_j} - \overrightarrow{v_i}\right)\cdot\overrightarrow{t_{ij}}$ the tangential velocity difference and the last coefficient lead to tangential direction $\overrightarrow{t_{ij}} = \left(-n_{ij}^2, n_{ij}^1\right)$. Note k and κ are large constants. Finally, we get:

$$\overrightarrow{f_{ij}} = \left\{A_i \exp\left[\left(r_{ij} - d_{ij}\right)/B_i\right] + kg\left(r_{ij} - d_{ij}\right)\right\}\overrightarrow{n_{ij}} + \kappa g\left(r_{ij} - d_{ij}\right)\Delta v_{ji}^t \overrightarrow{t_{ij}}$$

The interaction with the walls is treated in a similar way. For distance to wall W, direction, and tangential direction we utilized d_{iW}, $\overrightarrow{n_{iW}}$, $\overrightarrow{t_{iW}}$ to represent. The force can be calculated as follows:

$$\begin{aligned}\overrightarrow{f_{iW}} = &\left\{A_i \exp\left[\left(r_i - d_{iW}\right)/B_i\right] + kg\left(r_i - d_{iW}\right)\right\}\overrightarrow{n_{iW}} \\ &- \kappa g\left(r_i - d_{iW}\right)\left(\overrightarrow{v_i}\cdot\overrightarrow{t_{iW}}\right)\overrightarrow{t_{iW}}\end{aligned}$$

Therefore, a pedestrian can be modeled in the above way. We give a figure to represent our model and the final formulate is: $\overrightarrow{\xi}(t) = \overrightarrow{f_i^0} + \sum_{j\neq i}\overrightarrow{f_{ij}} + \sum_w \overrightarrow{f_{iw}}$

3.2 Multi-exit selection model

Pedestrians' exit-choosing process is one of the most critical decision-making processes for evacuation from multi-exit buildings. To aid this process, we have identified the factors that impact exit selection and developed an exit selection model. Our pedestrian evacuation model is based on the social force model and considers psychological factors that influence pedestrian speed and the effect of exits on pedestrians. Ultimately, we have established a multi-exit evacuation model by combining our exit selection and pedestrian evacuation models.

The pedestrian exit-choosing process involves objective factors such as the distance and width of exits. In real-life situations, pedestrians often do not choose the nearest exit. Instead, they choose safer exits, with lower crowd density, moderate distance, and wider width, after carefully considering all of their options. Three factors influence exit selection: distance, density, and exit width. Typically, pedestrians tend to choose the closest exit available to them. If the pedestrian density near the exit is high, they may look for alternate exits; however, if the density is low, they are unlikely to choose a different exit. In addition, Pedestrians are more likely to select exits that have a larger width.

Based on the above analysis of pedestrian density near an exit, we have established a threshold to differentiate between pedestrian densities affecting evacuation speed and those not. To accomplish this, we have created a pedestrian density function. The following are the rules for selecting an exit based on density factors.

1. When the pedestrian density near an exit is below the threshold, the probability of pedestrians choosing this exit is not impacted.
2. When the pedestrian density near an exit is higher than the threshold, pedestrians will opt for exits with lower density than the threshold, and the probability of pedestrians choosing an exit is inversely proportional to the density

We define pedestrian density as the ratio of the number of pedestrians in the exit effect region to the area of the exit effect region (as depicted in Fig. 2). The exit effect region is a semi-circle with a radius of λw_n, centered on the midpoint of the orthographic projection of exit n. In this region,

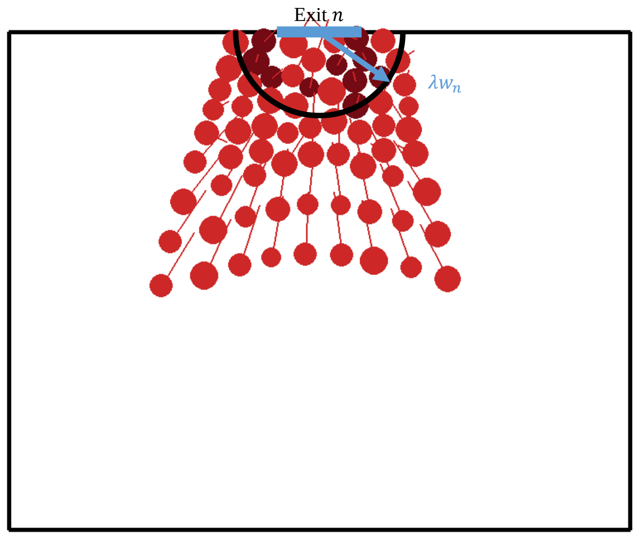

Figure 2: The enclosed area is the exit effect area. The blue bar on the black boundary indicates the exit while the blue arrow represents the radius of the affected area.

pedestrian behavior is affected by pedestrian density. The definition of pedestrian density d_n and the corresponding density function $g(d_n)$ is as follows:

$$d_n = \frac{2p}{\pi\left(\lambda w_n\right)^2}$$

$$g\left(d_n\right) = \begin{cases} d_n & d_n <= d_n^{\max} \\ d_n^{\max} & d_n > d_n^{\max} \end{cases}$$

In the above equation, d_n represents the pedestrian density near exit n, while p denotes the number of pedestrians in the exit effect area. The parameter λ serves as the regulatory factor in the exit effect area, while w_n represents the width of exit n. The threshold of pedestrian density near exit n is represented by $d_n^{\max}$.

The likelihood of pedestrians selecting an exit is determined using probability, which is influenced by three variables: distance from pedestrian to exit, pedestrian density near the exit, and width of the exit (as illustrated in Fig. 3). We provide the definition of the probability of exit selection for each variable separately as below. Finally, the multi-exit selection model is defined as a weighted sum of the probabilities calculated for all three variables.

$$P_n(r) = \frac{r_n^{-k_r}}{\sum_{n=1}^{N} r_n^{-k_r}}$$

$$P_n(d) = \frac{g\left(d_n\right)^{-k_d}}{\sum_{n=1}^{N} g\left(d_n\right)^{-k_d}}$$

$$P_n(w) = \frac{w_n^{k_w}}{\sum_{n=1}^{N} w_n^{k_w}}$$

In the above equation, N denotes the total number of exits available, while $P_n(r)$ represents the probability of a pedestrian selecting exit n based on distance factor r. Similarly, $P_n(d)$ represents the probability of a pedestrian choosing exit n based on pedestrian density d, while $P_n(w)$ denotes the probability of a pedestrian choosing exit n based on the width of the exit w. Here, r_n represents the distance between a pedestrian and exit n, and k_r, k_d, and k_w are constants that adjust the sensitivity of the distance, density, and width factors, respectively, on exit selection. The exponent value of these constants is adjusted to account for their influence on exit selection.

The weights for each factor are defined as follows:

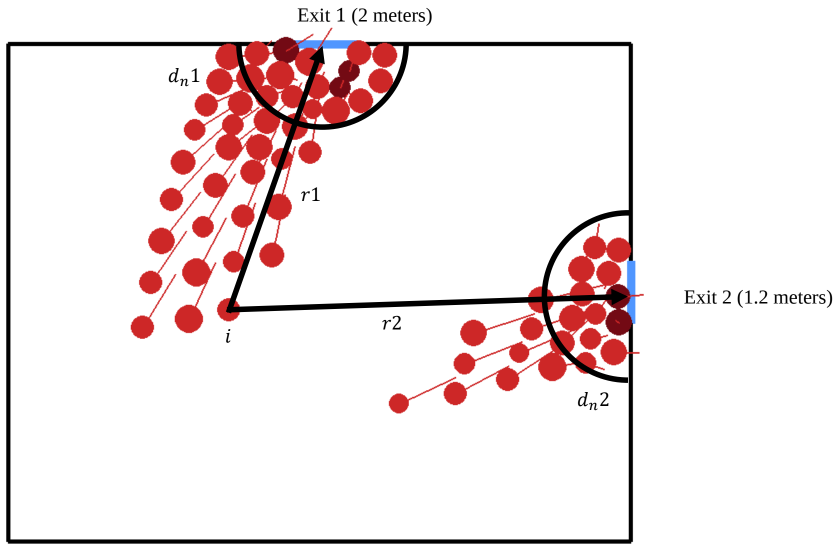

Figure 3: Three considerations for pedestrians when choosing exit.

$$\alpha = \frac{1}{N}\sum_{n=1}^{N}\left|1 - \frac{Nr_n}{R}\right|^{k_\alpha}$$

$$\beta = \frac{1}{N}\sum_{n=1}^{N}\left|1 - \frac{Ng\left(d_n\right)}{D}\right|^{k_\beta}$$

$$\gamma = \frac{1}{N}\sum_{n=1}^{N}\left|1 - \frac{Nw_n}{W}\right|^{k_\gamma}$$

In the above equation, α, β, and γ represent the weights of the distance, pedestrian density, and width factors, respectively. The terms $R = \sum_{n=1}^{N} r_n$, $D = \sum_{n=1}^{N} g(d_n)$, and $W = \sum_{n=1}^{N} w_n$ are the total distances, densities, and widths of all available exits, respectively. The constants k_α, k_β, and k_γ are the exponents of the distance, pedestrian density, and width weights, respectively, and are used to adjust the sensitivity of the corresponding weight on exit selection.

Therefore, a multi-exit selection model can be described below:

$$P_n = \frac{\alpha P_n(r) + \beta P_n(d) + \gamma P_n(w)}{\alpha + \beta + \gamma}$$

$$B_e = \operatorname{argmax}_{n\in\{1,2,3,4\}} P_n$$

where P_n is the probability that exit n (represent the index of exits) will be selected by the pedestrian, and B_e is the target exit for a pedestrian.

3.3 Revised multi-exit force model

After modeling the multi-exit model, we can generate the resultant force in this situation.

$$m_i\frac{d\overrightarrow{v_i}}{dt} = m_i\frac{v_i^0(t)\overrightarrow{e_i^0}(t) - \overrightarrow{v_i}(t)}{\tau_i} + \sum_{j\neq i}\overrightarrow{f_{ij}}(t) + \sum_{w}\overrightarrow{f_{iw}}(t) + \overrightarrow{f_{iB_e}}(t)$$

where

$$\overrightarrow{e_i^0}(t) = \frac{\overrightarrow{X_{B_e}} - \overrightarrow{X_i^t}}{d_{iB_e}}$$

note that $\overrightarrow{X_{B_e}}$ represents the desired location of B_e, d_{iB_e} is the distance from pedestrian i to B_e.

4 Simulation

4.1 Simulation Model

The Agent-based Model is implemented in an object-oriented fashion. The overall structure is designed such that each part is decoupled from the other, so adding or removing features without affecting the whole simulator's functioning is easy.

The structure is designed as the following:

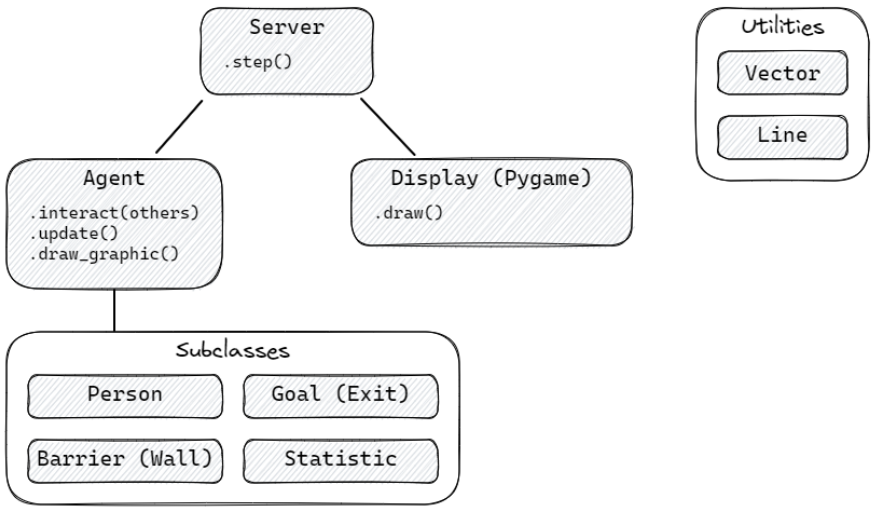

Figure 4: Simulator Model Structure

- `class Server`: The "manager" that holds all the agents in the simulation world, and calls their updating functions based on the `update_rate` (60 frames per second in our project).
- `class Agent`: The abstract class works as the base of all agents. An agent has the following methods:
  - `.interact(others)`: Get passed all the agents the server holds. Interactions like collisions with other agents (people or walls) or self-driving forces toward the exits should be handled here.
  - `.update()`: After all the agents' `interact()` are called, this will be called. Agents' states should be handled here, like position, velocity, acceleration, or statistics.
  - `.draw_graphic()`: This will be called if the display function is turned on. Should draw the graphics representing the agent if there are any.
- Subclasses of `Agent`: The real implementation of various agents. In this project, we have:
  - `Person`: The agent simulates a person's behavior in the evacuation. The social forces and exit selection described in previous sections are implemented here.
  - `Goal`: The exits or doors in evacuation scenarios. The doors have different widths and also keep track of the number of people within a certain distance.
  - `Barrier`: Basically the walls in our implementation. It could be extended to columns, furniture, or any other obstacles in a building.
  - `Statistic`: A virtual agent that keeps track of all the statistics we care about, like the time past, the people remaining in the room, the average velocities, etc.
- `class Display`: The class prepares the canvas to visualize everything: the room, the doors, the people, and the statistics. It will (be called by the server to) call every agent's `draw_graphic()` method every frame and tick the clock per the `update_rate` attribute of the server.

  When the tuning of configures is finished, and the batch of experiments is ready to go, the display will be turned off for faster simulation since we only need the statistics outputs.

- Some utility classes:
  - `Vector`: Used as both 2D vectors and 2D points. It provides functions like vector algebra, distance calculation between 2 points, dot product, cross product, and verifying if a point is inside a polygon.

    Positions, velocities, accelerations, and force calculations are all done with this class.
  - `Line`: It provides functions like length, extension, and closest point to a given point.

    Doors and walls are based on this class.

Our simulation models were developed with Python 3.10, and the simulation visualization was created with `pygame` library.

4.2 Verification

We verified our implementation by unit testing on each function followed by an end-to-end test. The elements of our simulator are well decoupled, and each is tested before being installed into the simulation pipeline. For the utility classes, we tested a group of calculations with them to ensure their correctness. For the people agents, we tested different parameters, like radius, mass, and velocity, and checked if the interaction (collision) between them as desired. For the wall agents, we tested their interaction with person agents of different velocities and masses. For the goal agents, we drew their detection range and checked their counting of the person agents. For the server, we tested its performance with different update rates and settings.

5 Experimental results and validation

5.1 Parameter settings

As we verify the simulation model, we proceed to perform experiments and analyze the results. For the design of the experiment, we start with a room with a length of 15m and width of 12m, radius r of people with distribution Unif$(0.25, 0.35)$, and mass $m = 20 + 1850r^3$ referring [HFV00]. We first examine our model with fixed initial positions and a fixed number of people for different scenarios by changing the number of doors (1, 2, 3, and 4), the width of doors (1.5m and 3m [HR21]), and desired speeds (1.5ms^{-1}, and $v_0 \sim \mathcal{N}(1.34, 0.26^2)$, the Gaussian distribution with mean 1.34ms^{-1} and standard deviation 0.26ms^{-1} [HM95]) (e.g. Figure 5). Note the strategy of the multi-exit selection model we used here considers all three factors, i.e., distance, density, and width, as Zhang et al. [ZHJ$^+$21] shows that considering all three factors is most efficient and makes the simulation closer to reality.

Since in the real world, the initial positions of people are unlikely to be the uniform distribution scenario, we also test the model for different numbers of people and different initial positions in distributions like a normal distribution, uniform distribution, random distribution, and beta distribution Table 2.

Though from Table 1 we can see that changing the desired speed to Gaussian distribution causes a delay in time, the results are closer to the real-world cases than the desired speed of 1.5ms^{-1}. Additionally, our experiment results suggest that the spatial distribution of people in a room significantly impacts evacuation time. Specifically, when people are normally distributed, they tend to evacuate more quickly than when clustered together or dispersed throughout the room. This is likely because normal distributions provide more even and efficient pathways for people to move through. However, we observed that as the number of people in the room increased, the overall evacuation time also increased. This suggests that there may be a threshold beyond which adding more people to a space results in diminishing returns in terms of evacuation efficiency. Moreover, as more people are added, the average speed at which each individual moves decreases, likely due to increased congestion and interference from other evacuees.

According to the previous settings, we finalize our parameters and proposed more realistic experiments with a room of dimension 12m×15m and a capacity of 50-200 seated, 300 standing referring to the dimension and capacity of Georgia Tech The Atrium. For the mass, we have $m = 20 + 1850r^3$ with r as the radius of people follows the uniform distribution Unif$(0.25, 0.35)$ which gives the range between $[48.9, 99.3]$kg. For the desired speed v_0, according to Helbing et al. [HFV00], we chose $v_0 \leq 1.5$ms^{-1} to be the initial setting under nervous condition and $v_0 \sim \mathcal{N}(1.34, 0.26^2)$ [HM95] in previous tests,

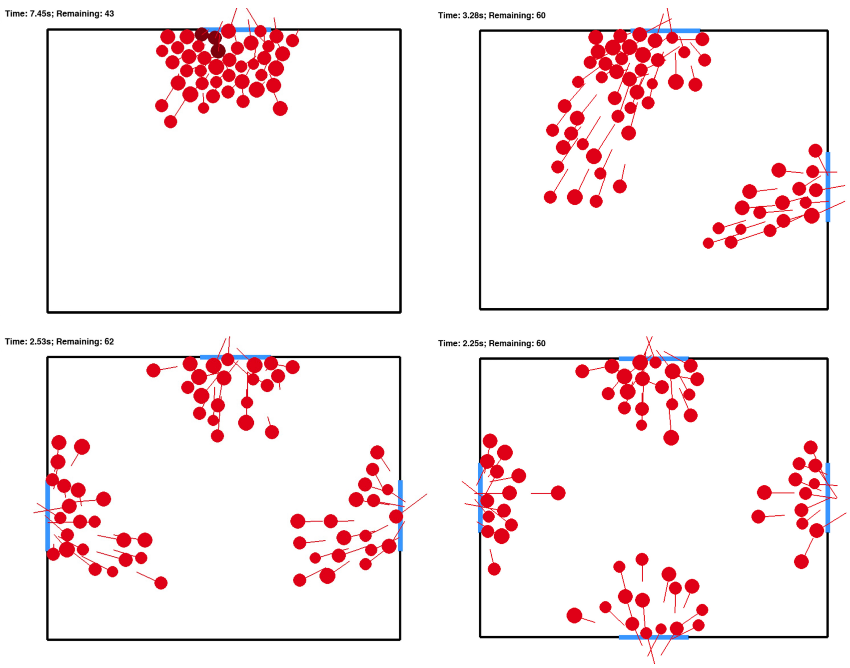

Figure 5: Example simulations for 3m doors with desired speed $v_0 \sim \mathcal{N}(1.34, 0.26^2)$

Number of doors	Door Width	Desired speed	Mean	Time
1	1.5	1.5	1.154	37
	1.5	$v_0 \sim \mathcal{N}(1.34, 0.26^2)$	0.607	41.867
	3	1.5	0.970	12.850
	3	$v_0 \sim \mathcal{N}(1.34, 0.26^2)$	1.080	14.467
2	1.5	1.5ms^{-1}	1.166	26.283
	1.5	$v_0 \sim \mathcal{N}(1.34, 0.26^2)$	0.774	28.567
	3	1.5ms^{-1}	1.23	10.95
	3	$v_0 \sim \mathcal{N}(1.34, 0.26^2)$	1.153	11.633
3	1.5	1.5ms^{-1}	1.305	14.85
	1.5	$v_0 \sim \mathcal{N}(1.34, 0.26^2)$	0.670	15.183
	3	1.5ms^{-1}	1.359	6.133
	3	$v_0 \sim \mathcal{N}(1.34, 0.26^2)$	1.367	7.117
4	1.5	1.5ms^{-1}	0.766	12.183
	1.5	$v_0 \sim \mathcal{N}(1.34, 0.26^2)$	0.685	12.250
	3	1.5ms^{-1}	1.297	5.417
	3	$v_0 \sim \mathcal{N}(1.34, 0.26^2)$	0.983	6.300

Table 1: Overall Descriptive Statistics of All Door Conditions

Number of People	Distribution of People	Distribution Parameter	Time
30	Normal	$\mu_x = 0.5, \mu_y = 0.5, \sigma = 0.1$	7.13
	Random	seed=1	8.80
	Uniform	min=0, max=1	10.38
	Beta	a=0.5, b=0.5	9.71
50	Normal	$\mu_x = 0.5, \mu_y = 0.5, \sigma = 0.1$	9.67
	Random	seed=1	9.25
	Uniform	min=0, max=1	10.63
	Beta	a=0.5, b=0.5	10.28
100	Normal	$\mu_x = 0.5, \mu_y = 0.5, \sigma = 0.1$	11.93
	Random	seed=1	12.4
	Uniform	min=0, max=1	12.53
	Beta	a=0.5, b=0.5	11.88
200	Normal	$\mu_x = 0.5, \mu_y = 0.5, \sigma = 0.1$	20.28
	Random	seed=1	19.92
	Uniform	min=0, max=1	20.65
	Beta	a=0.5, b=0.5	24.81

Table 2: Overall Escape Time with Different Distribution of People

and as discussed in the previous section, we choose $v_0 \sim \mathcal{N}(1.34, 0.26^2)$ to be the desired speed in the experiment. And for different scenarios of the initial distribution of people, the times are similar as the number of people increases. Hence for the final setting, we use random distribution for the initial positions of people.

5.2 Experimental results

For the final experiment, we fix four parameters and two variables in the room of dimension 12m×15m

- Desired velocity: $v_0 \sim \mathcal{N}(1.34, 0.26^2)$
- Mass: $m = 20 + 1850r^3$ with $r \sim$Unif$(0.25, 0.35)$
- Initial position: Random distribution
- Multi-Exit model factor: Distance & Density & Width
- Number of doors: 1, 2, 3, 4
- Width of doors: 1.5m, 3m

For each simulation, we collect the average and standard deviation of the evacuation time for 300 people in different scenarios. Table 3 shows the overall descriptive statistics of two dependent variables, EF (evacuation flow) and ET (evacuation time) with different exit conditions. For comparisons of evacuation time, we run a test of 50 people with a door width of 3m due to the running time. Table 4 shows the comparisons of times between different numbers of doors with a 95% confidence interval.

5.3 Validation

From figure 6, we can see that the slope of evacuation considering the width of doors is almost the same, i.e., the number of people evacuated per time as the evacuation efficiency is similar. It is clear that in our model, as the width of the door increase, it takes less time to evacuate. Compared to the model proposed by Hessanpour et al. [HR21] which shows it is unnecessary to have more than three doors and the width of doors does not affect evacuation times for that, our model also indicates it may be unnecessary to have more than three doors but there exists a great difference between total evacuation times for 300 people of door width 1.5m and 3m as shown in table 3. For the impact of the number of doors, table 4 shows there is no significant difference between 1 and 2 doors or 3 and 4 doors with 3m doors for evacuating 50 people. As the number of people increases, we can derive

Number of Doors	Door Width	Mean(EF)	Std(EF)	Time
1	1.5	0.300	0.244	60.267
	3	0.589	0.262	25.833
2	1.5	0.449	0.340	42.117
	3	0.764	0.239	18.283
3	1.5	0.490	0.311	34.217
	3	0.797	0.262	12.883
4	1.5	0.568	0.333	27.533
	3	0.827	0.275	10.050

Table 3: Overall Descriptive Statistics of All Door Conditions

Number of Doors	Mean(ET)	Std(ET)	Lower Bound	Upper Bound
1	15.914	1.849	12.159	19.668
2	12.994	2.657	7.621	18.368
3	10.510	2.633	5.165	15.855
4	8.239	2.142	3.891	12.587

Table 4: Statistics of Evacuation Time between Different Numbers of Doors

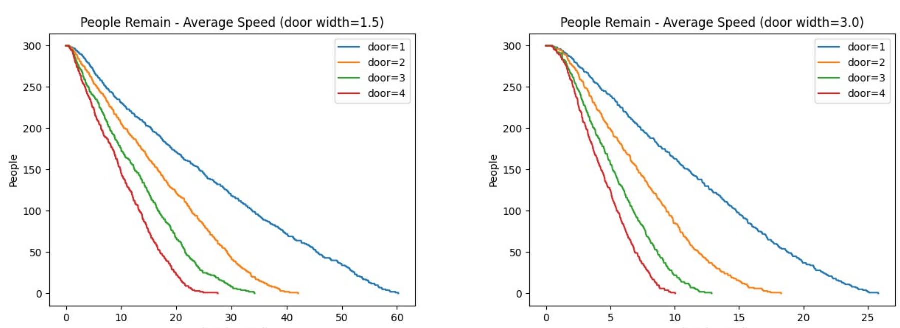

Figure 6: Number of evacuated persons vs Evacuation Time under four situations

that it is not necessary to have 4 doors compared to 3 doors. A similar conclusion is also obtained in previous studies [HR21] and [CYS16] as the number of people increases. With full consideration of factors of distance, density, and width, the results of the experiments with evacuation flows and times in different settings show our model is valid and can improve evacuation efficiency by changing the exit conditions.

6 Conclusion

In this project, we incorporate the social force-based model (SFM) with a multi-exit selection model considering all three factors (distance, density, and width) under different exit conditions. Three impact factors are assigned weights according to people's behavior under personal goals and mentality. Then we combine our probability and weight and finalize our model into an equation. Our result shows that the physical attributes of the social force and exit area conditions could affect the evacuation process such that having 3 doors with a width of 3m would be enough for evacuation in an atrium or large lecture hall.

Our agent-based model has an effective structure that supports flexible adding and removing features. Therefore, we adding exits and changing distribution of people to do some of ablation studies to deliver our final conclusions in last paragraph. However, the agent-based model is computationally intensive, so Python is the best choice. Implementing this simulator with another language that performs better in efficiency, such as C++ or Java, would be ideal. Moreover, some researches also pointed out that if we do not assign every pedestrian with force model and exit choosing model, instead. Specify the leader model that people intends to follow, there will be other new features and will be more realistic.

For the desired speed, according to [HM10], the speed of evacuees follows normal distribution such that the average speed of men is slightly larger than that of women. In the future, we may adjust the simulation model such that the desired speed of each evacuee follows the gender which may be distinguished by the mass. And [HR21] also proposed that the speed can be between $[1.1, 1.5]\text{ms}^{-1}$ in normal and 2.5ms^{-1} in emergency. For the evacuation time in different situations, we only tested 50 people with a door width of 3m by changing the number of doors. For future work, adding the number of evacuees may give more accurate results.

7 Division of labor

- Zihong Hao: Code implementation, visualization, and report.
- Qilin Li: Experiment, analysis, and report.
- Hanzhang Liu: Modeling, simulation settings, and report.
- Chen Lin: Experiment, analysis, and report.
- Mian Wu: Modeling, simulation settings, and report.

References

[BCGV14] Stefania Bandini, Luca Crociani, Andrea Gorrini, and Giuseppe Vizzari. An agent-based model of pedestrian dynamics considering groups: A real world case study. In *17th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pages 572–577. IEEE, 2014.

[BG12] Bert Blocken and Carlo Gualtieri. Ten iterative steps for model development and evaluation applied to computational fluid dynamics for environmental fluid mechanics. *Environmental Modelling & Software*, 33:1–22, 2012.

[CLQ96] Bastien Chopard, Pascal O Luthi, and Pierre-Antoine Queloz. Cellular automata model of car traffic in a two-dimensional street network. *Journal of Physics A: Mathematical and General*, 29(10):2325, 1996.

[CYS16] Yi Chen, Rui Yang, and Shi Shen. Impact study on mass evacuation in urban underground passages. *International Journal of Engineering and Technology*, 8:222–226, 03 2016.

[HFV00] Dirk Helbing, Illés Farkas, and Tamas Vicsek. Simulating dynamical features of escape panic. *Nature*, 407(6803):487–490, 2000.

[HM95] Dirk Helbing and Peter Molnar. Social force model for pedestrian dynamics. *Physical review E*, 51(5):4282, 1995.

[HM10] Canbin Huang and Wanjing Ma. A statistical analysis of pedestrian speed on signalized intersection crosswalk. *Tenth International Conference of Chinese Transportation Professionals (ICCTP)*, August 2010.

[HR21] Sajjad Hassanpour and Amir Abbas Rassafi. Agent-based simulation for pedestrian evacuation behaviour using the affordance concept. *KSCE Journal of Civil Engineering*, 25(4):1433–1445, 2021.

[LLZ+18] Hong Liu, Baoxi Liu, Hao Zhang, Liang Li, Xin Qin, and Guijuan Zhang. Crowd evacuation simulation approach based on navigation knowledge and two-layer control mechanism. *Information Sciences*, 436:247–267, 2018.

[PM08] Nuria Pelechano and Ali Malkawi. Evacuation simulation models: Challenges in modeling high rise building evacuation with cellular automata approaches. *Automation in construction*, 17(4):377–385, 2008.

[RHE11] Tyng-Rong Roan, Muki Haklay, and Claire Ellul. Modified navigation algorithms in agent-based modelling for fire evacuation simulation. In *11th International Conference on GeoComputation, London*, 2011.

[ZHJ+21] Dezhen Zhang, Gaoyue Huang, Chengtao Ji, Huiying Liu, and Ying Tang. Pedestrian evacuation modeling and simulation in multi-exit scenarios. *Physica A: Statistical Mechanics and its Applications*, 582:126272, 2021.