COVID-19 social distancing simulation: a supermarket case study

Joris Hilberink (1009790), Jiaxin Dai (1602543) & Ruixin Tang (1243969)
Group 14b

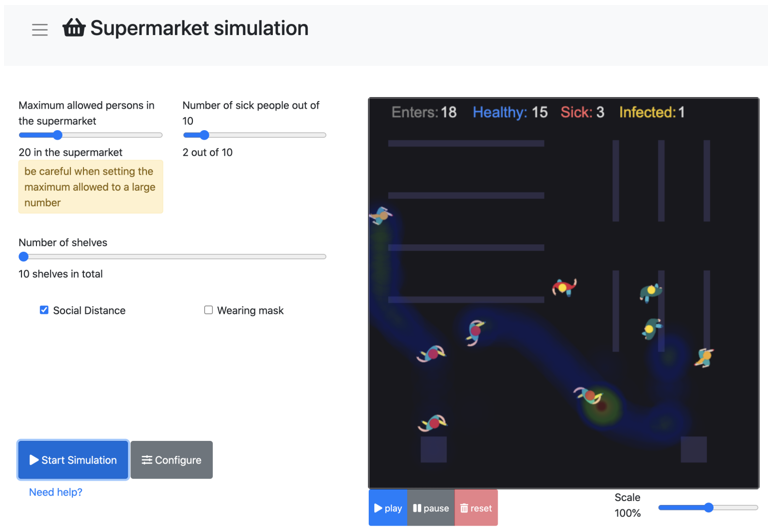

Fig. 1: An overview of the COVID-19 supermarket case visualization tool.

Abstract—There are several simulations for people to represent and calculate the infection spread of COVID-19. However, the spread of COVID-19 and the effectiveness of different measures in specific scenarios are not visually clear in those simulations. This report describes a tool built to provide a visualization for shop owners and policymakers to show the spread and the influence of COVID-19 in a supermarket. Four essential measurements were extracted as domain-specific tasks for a better understanding of COVID-19 spread in the visualization tool. Possible aspects including the configuration of a supermarket shopping rule, different measures, and the layout of the supermarket are taken into considerations. The results of these aspects are shown as user cases and evaluated based on the comparison of the infected rate by limiting the simulation time. It turns out that wearing masks is the most necessary measure in the supermarket and limited customers allowed combined with social distance can also make a difference in the infection rate. In addition, limitation on the supermarket open time can decrease virus transmission as well.

Index Terms—COVID-19, supermarket simulation, pandemic measures, virus, social distancing, visualization. . .

1 Introduction

The COVID-19 pandemic has spread around the world over the past year. This virus can be transmitted from a person to another by airborne droplets. Apart from vaccinations, the means in terms of combating the disease is a set of measures including wearing masks, quarantining, lock-downs, evening clocks and more. Among all the measures implemented, social distancing has proven to be a successful measure since COVID-19 spreads mainly among people who are in close contact for a prolonged period of time [3]. However, the impact of the measures is hard to understand for people, even for the persons who set these measures as well, because new insights are continually developed. It is advisable to simulate and assess these prevention measures to make people understand how the virus is transmitted in a certain space in real time. In this sense, a more optimal strategy can be recommended with less harm or misunderstandings.

To give people a better sense of why certain measures are taken and to minimize the potential risks, an intuitive way of visualizing the COVID-19 transmission is needed. Based on the given supermarket simulation model [6], our goal is to create an interactive visualization that can serve as a decision support tool for policy makers or, in this case, the grocery store owner. In this concept report, we explain how we aimed to create an interactive visualization to help supermarkets and other stakeholders with the implementation of effective measures to reduce the risk of transmissions in the supermarket environment.

1.1 Problem Description

Although the simulation serves as a sufficient solution for calculating the infectious possibilities by close contact or via inhalation of virus-containing aerosols, the current application is still limited. It is relatively hard for people to interpret it and apply to practical uses without a visual way to understand it. First, most visualizations that can be found are created in a general way to demonstrate global, national or regional statistics. A tool for smaller areas are relatively limited. Second, models built about COVID-19 are hard to interpret for normal people when it comes to particles, equations, and professional terms. Finally, the current models are often not interactive. This makes it difficult to evaluate and compare different scenarios.

Along these lines, it is necessary to create a tool that can input different parameters and visualize the process. In this project, our goal is to create an interactive visualization which could serve as a decision support tool for policy makers or in this case the grocery store owner.

2 Task Analysis

2.1 Domain Specific Tasks

Our tool mainly targets at helping the supermarket owners to gain insights into the impact of possible measures to take, allowing them to make comparisons among different regulations, understand effects of enforced measures, and carry on further analysis of the decision making process. It also provides the public a straightforward impression on the spread of COVID-19 in a supermarket. Specifically, a set of related tasks are brought up:

- How does COVID-19 possibly spread in a supermarket without any measures?
- How does the infection rate in a supermarket change based on the selected measures?
- How does the layout of a supermarket impact the infection probability?
- Which measure should a supermarket owner implement to minimize the possibility of infection?

These tasks require us to present the model and data properly, which allows the users to generate hypotheses about the differences over all the measures that can be taken by a supermarket. A comparison will be needed to show the deviations of the infection rate between various measures. Focusing on these tasks, we organized and selected a proper design to encode and visualize the results.

2.2 Task Abstraction: Why

Using the simulation of the possible COVID-19 transmission in a supermarket, our proposed tasks can be transformed into visualization tasks. Our first task is mainly about presenting, to firstly provide an overview of the the transmission and get a vivid understanding of the risk of spreading without any measures. The second tasks specifically focus on the effect of a single regulation or measure. We plan to focus on how different properties, such as how the number of people and patients would affect the visual output in our model. The third task is creating a real-time simulation display in a layout that can be set by users. We wanted to make sure that the supermarket can be edited and customized by users, creating a space that is highly interactive and consistent with reality, thus enhancing the efficiency of use. The last task will be done by providing our conclusion from the analysis. Hence, we are able to provide a tool for policy makers and shop owners to make decisions to prevent virus spreading.

To meet the demands of completing a possible transmission model, we set several requirements for implementation in order to testify all the tasks we mentioned.

- Interface: the task is to create a central web page including a parameter panel, a simulation window, statistics output and interactive buttons to control the progress.
- Components: the task is to actualize the environment, the consumers, the uninfected people, the COVID particles, and the COVID measures including social distance, masks and max number of people in the supermarket.
- Properties: we would like to show the time dimension, the default sets, the scale of the layouts along with shelves and walls, and the statistics while running.
- Interactions: the ideal version allows users to edit starting values, select measures, play and pause at any time, and design the layout of the supermarket.

To sum up, these requirements are the premise to demonstrate a real-time simulation to be seen through time. We also intended to design a platform that enables users to change, create and explore by their preference. These requirements are set to testify our outcomes in a quantitative and effective way when it is finished.

3 Literature Discussion

3.1 Numerical simulations by inhalation indoors

In models given in [6], they study mostly individuals walking at the regular speed indoors, specifically, a generic supermarket. The floor plan is fixed and based on publicly available data on a typical Finnish supermarket. They generated some statistics on the probability density function of the number of inhaled particles. Considering the number of supermarket visits daily, the number of exposed individuals is significantly large in populations.

However, there are some limitations to this supermarket simulation model. First, the output of the model is relatively too numerical. They have a separate output channel using the print function in python, which can only be seen in the console. Secondly, they do not have an interface that a user can easily understand. An example of the image output can be viewed in Figure 2. Since no UI is provided, these image outputs are saved in the file directory step by step, resulting in thousands of pictures without any animation of the simulation.

3.2 COVID-19 transmission study in Medical Faculty

As stated in a recent paper published in Nature, the moment a person breathes or talks, sneezes, or coughs, a fine spray of liquid particles takes flight. Aerosols could prove to be the most important transmission vehicle. A normal mask can block half of the inhaled aerosols and almost 80% of exhaled aerosols measuring 2 µm across [5].

Figure 3 shows the infection possibility under different situations. We find this kind of picture very informative and convey true suggestions to the public. However, these results are done with experiments by medical faculties in a much larger population scenario. It may not be the case when fitting with a different condition. These are only shown in pictures with statistics (the

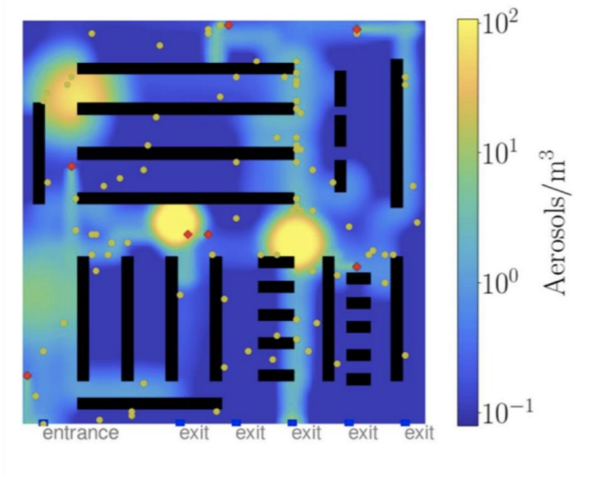

Fig. 2: An example image output of the model in [6]

probability that COVID-19 will spread), but seldom in animations or dynamic visualization tools that can fit into a certain situation.

4 Visualization and Interaction Design: How

Focusing on our tasks, we will implement the visualization techniques based on the model outputs and interaction designs to help users carry on analysis.

4.1 Graphic information of the spread of COVID-19 in a supermarket

To present a dynamic graphic visualization of COVID-19 spread in a supermarket, we use Matter.js, a light-weighted physical engine, as an external library. The simulation result of the COVID-19 spread in a supermarket is presented with 2D animation, with one step of changing each frame.

The state of a customer is encoded as the color of the 'person' object, with red representing the original COVID-19 patients, blue representing the healthy persons, and yellow representing the possible infected person. This can be viewed when the simulation is paused, which gives a intermediate state of the whole simulation. This visually shows the user if everyone in the supermarket is currently sick or a possible COVID-19 carrier. Furthermore, we also implemented sprites representing different persons and to show who is infected. The sprites also visually show certain measures, such as social distancing with shopping carts. Each 'person' object has its own target array, simulating the shopping list of a customer. The shelves are static and can be viewed as obstacles for the customers.

The whole window with shelves and the customers gives a vivid overview of the current state inside a supermarket, including the possible contact and infections between customers. It illustrates information, such as the layout of the supermarket and the current position of the customers in a graphical way.

4.2 Numerical evolution of the COVID-19 cases in a supermarket

While graphical simulations are easy to understand, we still need statistics to truly convince supermarket owners that the measures they are taking are working or not working. Various indicators of the number of people in the supermarket are recorded and

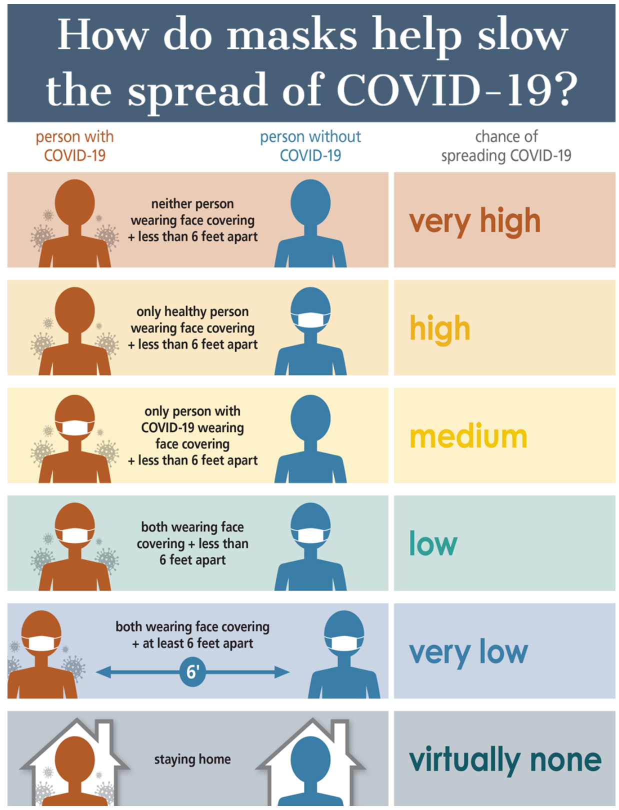

Fig. 3: Infection possibility decreases as the measures are taken [4]

updated in real-time, such as the total number of people in the supermarket, the number of potential infections in the supermarket, and the number of healthy people in the supermarket. At the completion of the simulation, the program will give the current simulated infection rate based on the given parameters. Depending on the infection rate, we can prove or disprove our assumptions on the effectiveness of these COVID-19 measures in supermarket.

4.3 The spread of particles

By breathing and possibly coughing, the customers who are carrying COVID-19 will release particles that may contain the COVID-19 virus with a chance of infecting other people around, according to the simulation model we build. An additional heatmap layer is added on top of the simulation window. The density of the particles within the supermarket is encoded as the color in the heatmap layer. The transition from transparent to red indicates the density of the particles from none to very high. This will notify users of the possible unreasonable design of the layout. For example, if the walking spaces between shelves are narrow, customers may have more chance to contact each other, which will result in a more reddish area in the heatmap layer.

4.4 Possible COVID-19 measures

In reality, many measures related to COVID-19 are announced and carried out by either the government or individual industries. The statistics are updated every day. However, there is no clear visualization on how effective these measures are. With the implementation of multiple COVID-19 measures, users are able to apply or remove them from the simulation. Such measures include the mask usage, the social distance enforcement, and a limitation on maximum number of people in the supermarket.

The settings and parameters that are passed to the model will be adapted to these measures accordingly and in real-time in the appropriate time dimension. For example, if the mask measure is applied, the COVID-19 particles spread should be reduced.

4.5 Supermarket layout designer

The supermarket layout will be designed using the top view, with shelves and free moving space, together with a clear indication of the entrance and the exit. For normal users, who would only like to see the possible spread of the virus in a indoor area, default supermarket layouts will be provided in advance.

An additional design panel will be created for a more specific and detailed design of the supermarket. This function is more suitable for the supermarket owner who would like to make decisions in a certain shop. It allows users to create a similar simulation environment to their actual supermarket, for giving realistic and accurate simulation results and to facilitate practical decisions based on the measures.

For this to work properly, a scale is needed such that the person recreating the layout can make an accurate representation of the actual supermarket. Dealing with this problem, we created a grid overlay in the design editor that represents a certain real-world size (i.e. one grid area is 1m^2). This grid resizes accordingly when the layout is scaled, such that it can be used to create accurately sized supermarket layouts.

5 Realization

Our visualization tool is built using JavaScript. It is a powerful language for front-end visualization, which can be easily combined and embedded in HTML. Other advantages are that it is compatible with most browsers and that it does not need to be compiled. It can easily be extended with libraries, such that we do not need to re-invent the wheel, and we use Matter-js and heatmap.js as helper libraries for a more efficient implementation of our visualization tool. Here is how we meet requirements we set before to complete a possible transmission model.

- Interface: We create a central web page including a parameter panel in the left and a simulation window in the right. We draw statistics from the original model and show them in the simulation window. Some interactive buttons can be used to start, pause and reset.
- Components: We actualized the environment, the consumers, the uninfected people, the COVID particles, and the COVID measures including social distance, masks and max number of people in the supermarket. All the components can be seen in the visualization window.
- Properties: We set the time dimension, the default sets, the scale of the layouts along with shelves and walls, and the statistics while running.
- Interactions: Users can input starting values, select measures, play and pause at any time. The layout of the supermarket can be changed totally by the user.

Overall, we meet our basic requirements of interface, components, properties and interactions. In our implementation, the layout designer and the main simulation window have a panel for changing parameters. Detailed functionality will be discussed in the following sections.

5.1 Simulation Model

In the simulation model, we model up to 100 individuals in a 2D area of size 600×600(in default scale) walking at a fixed speed for a fixed duration of 3 hours. The time was scaled that every second in the simulation is roughly the same as one minute in the real supermarket.

We investigated that a speaking person emits at the rate of 0.01 unit particles while in a coughing scenario 0.6 unit particles per cough are assumed. A cough or a sneeze is associated with a rapid burst of exhaled air where all the mass and momentum are released into the surroundings. As stated in [6], the spreading of the particles of a cough or a sneeze is equal to the normal 1-minute spreading.

The A-star path-finding algorithm is implemented by an external library called JavaScript-astar.js [1]. Every customer will have a random shopping list assigned when entering the supermarket. The list contains at least two items that are randomly generated on the shelves. Since the A-star algorithm is computationally expensive, we implemented several workers to calculate the optimal path for each customer. A worker is the web-equivalent of a background thread. The movement update function works as follows:

If no path is available to the person, it stays in the current position and calculates the path to the next target using the a-star method of the worker. If a path is found, the velocity vector for the next step is set to the next position in the path. If the path is finished, it is checked if the person reached the goal. If this is the case, the algorithm resets for the new goal. Otherwise, a new path is calculated. Furthermore, the algorithm check if there are other persons nearby. If it is the case, it checks if this person is behind a wall or not. If this is not the case, the two persons are opposed from each other by means of a rotation of their original vectors.

Where possible, the simulation uses a worker for every person that is currently active in the store. If no worker(s) is available, the a-star path is calculated on the 'main' thread, but this means that the simulation slows down significantly.

One improvement of our model compared to the original Monte Carlo B model is that we simulate the potential infection between persons. As people are contacting each other, it is possible that a healthy person will be infected by a passing-by coughing sick customer. The probability that one is potentially infected by a sick person depends on three factors, the distance between the healthy person and the sick person, the time the healthy person stays in the range of infection, and the original infection rate of the COVID-19 virus.

5.2 Visualization Window

The visualization window contains multiple features that help users better understand the spreading of the COVID-19 in an indoor place.

Firstly, the whole window represents the environment of the supermarket. In the default setting, the boundaries (the walls) of the supermarket are set to coincide with the window boundaries. In the designed layout setting, the user can decide the shape of their supermarket and the wall is presented as white lines in the window. Two box-like squares represent the entrance and exit. Shelves are represented by gray rectangles with a certain width and length. Goods are invisible since they should be on the shelves, but each of them has a certain location for path-finding algorithm to work properly. The customers are animated as walking persons in the supermarket.

On top of the canvas, there is a separate layer for visualizing the particles spreading of a COVID-19 patient. Heatmap.js [2] will be used as an external library to create a heatmap layer above the simulation windows. In order to keep the program working properly with more simulated customers, we update the heatmap only when it finds a different maximum value in the grid that saves the level of particles. The red indicates that within this area, there is a higher chance that a healthy person will get infected. Lastly, there is a fixed place on the top of the window for the statistics. Encoded with color, they clearly show the current state of the supermarket at a specific time point.

5.3 User Interface Design

The interface can be seen as figure 4. To the left, there is a configuration panel that provides users with the chance to change the number of sick persons (out of 10), total customers, and the shelves. In the right side, it is the visualization window to show the real-time situation of how COVID-19 spread in the given condition. The scale is set to be 100% as default. There is a real-time statistic showing how many people is getting infected in the top of the window. User can click ”start simulation” to see a dynamic simulation, and can also click ”pause” and ”reset” to control the progress. After the simulation is finished, there is a infection rate automatically generated and shown.

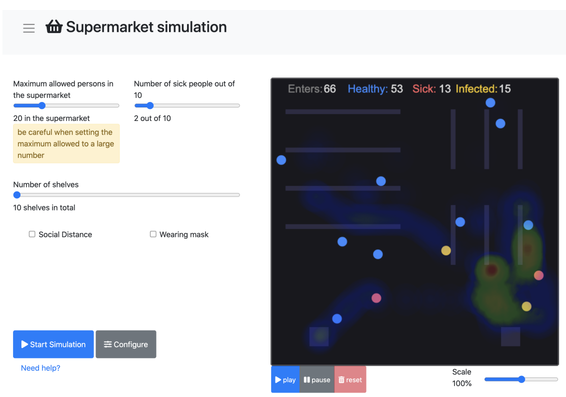

Fig. 4: Main interface of the simulation

By clicking the configure button, the layout of the supermarket can be personalized by the user. The drawing pad and the grid can be scaled accordingly and the length and the width of the shelf will be shown as an indication while drawing. This helps users to draw a more specific and accurate layout that can be well-simulated by our model. Users can also draw the wall (boundaries) of their supermarket to get a similar shape. By double-clicking, the wall will be finished automatically by connecting the starting point with the clicking position. The entrance and the exit are necessary to add to the environment. Please note that the given shelf and wall are not suitable for a successful simulation. In the configurator, a closed wall and an entrance and exit should be available.

6 Use Case and Evaluation

We would like to demonstrate how the simulation can be used to perform the analysis by our potential users, including policy makers and shop owners, to help them make reasonable decisions in pandemic time. In this sense, we show examples on how can the questions posed in the problem description be answered by using the tool.

How does COVID-19 possibly spread in a supermarket without any measures?

In order to evaluate the question, it is necessary to demonstrate the visualization in a way that users want to see how the infections happen and spread in the supermarket. This can be done by using parameter to change the amount of people in the supermarket, the people who is originally infected to show how the statistics fluctuated. This can be done by setting the maximum allowed persons and the number of sick people at the beginning. In this question, we assume that the layout of the supermarket is as the default and observe the difference between infection rate by different amount of people. By setting this, users could take the current infection rate on the basis of public data to make predictions. This would give them a clear vision of how the COVID-19 spreads without any measures and how the maximum number of customers affect the infection.

We could set an example to observe. When 25 people come to the supermarket with 5 people infected and when 10 people entered with 2 people infected, the simulation shows as Figure 5. The original infection rate is 20%. The final infection rate is 34% and 25% respectively. For 25 people allowed, we see a rise of 14% while for 10 people allowed, the rate increase by 5%. It is clear that if more people are allowed in a restricted area without any measures taken, it is crucial to limit the mount of customers allowed. We could conclude that with the same infection rate at the beginning, more people would be infected if the maximum people allowed increases.

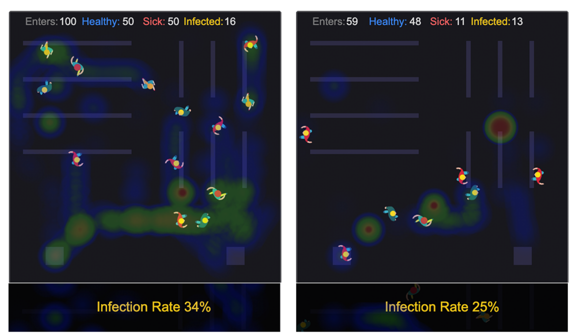

Fig. 5: Comparison of 25 people and 10 people allowed with infection rate=20%. Left:25 people; right:10 people.

It is also possible to compare the same amount of people under different original infection rates to give the user a comprehensive understanding. For example, we could set that 20 people can enter the supermarket with 5 or 1 sick person, presented in the Figure 6. After 1 min simulation, the rate is 30% for 5 infected people entered and 16% for only 1 infected person entered. Not surprisingly, the original infection rate is the most crucial factor that determines the outcomes. We can also conclude that even when very few infected people entered the supermarket, being exposed totally would actually cause an infection in a short period of time. This could give user a direct impression on how COVID-19 possibly spreads.

Furthermore, as we transformed the original model from

Fig. 6: Comparison of 5 and 1 infected people entered when 20 people allowed. Left:5 infected; right:1 infected.

python to JavaScript, we also made it possible to pause the model at any time to see how the statistics change over time. This model can run 1 minute currently, which approximately equals 2 hours real-time. When there are 20 people allowed with 2 infected people, we compared the statistics at 45s, 90s, 135s and 180s. We also draw a line chart to get a of how the infection change from time to time, which is shown in Figure 7

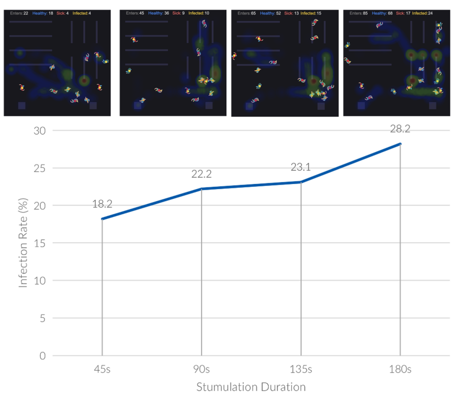

Fig. 7: The infection rate change through time when 20 people allowed and 2 infected people in the supermarket

How does the infection rate in a supermarket change based on the selected measures?

To answer this question, we applied 3 different measures that are actually being implemented in real life. First is the limitation of people allowedwhich is already shown in the last paragraph. Second, we apply the social distance rule to our model. The rule is that people should maintain 1.5 meters in the supermarket to each other. Because the shopping carts are actually within the distance of 1.5 meters, it is unnecessary take into account. The third measure is wearing a mask which is mandatory for every supermarket in the real life. If a person wears a mask, the particles of the breathing have a limited spreading distance, which could be seen in the heat map. All measures could be changed in the parameters by choosing if it should be applied or not. If we allow 20 people in the supermarket and 2 people are originally infected, we could make a comparison between different measures. The first comparison is the people applying social distance or not, which could be seen in Figure 8. The result show a decrease in infection rate from 41% with no measures to 21% with only the social distance measure. Then the outcome of applying mask or not can be seen in Figure 9. Similarly, we can witness a drop in infection rate from 41% with no measures to 0% with only applying the mask measure. In conclusion, those measures are significantly effective to prevent virus spreading in the simulation. This would also address the importance and raises awareness of implementing measures to users.

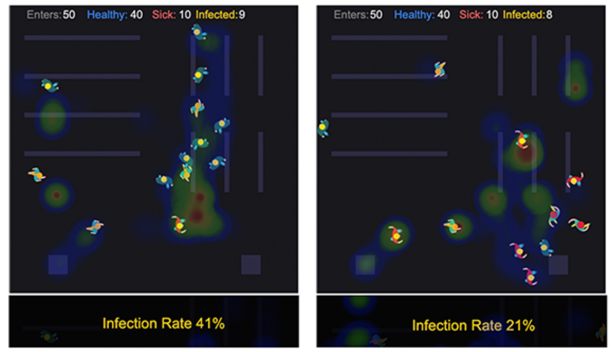

Fig. 8: Comparison of 20 and 2 infected people entered. Left: no measure taken; right:Social distance.

How does the layout of a supermarket impact the infection probability?

In the perspective of a shop owner, it is handy and convenient for them to apply exactly the identical layout to make the best prediction. That is why we did our best to create a custom-made layout to show the impact. To achieve this goal, users can either apply 3 different level of crowdedness of shelves or use "configure" as a helpful tool. Considering the time and effort to spend on the simulation, we would like users to have a fast and easy way to instinctively gain the results. There are three options to choose from before you start the simulation which can be seen in Figure 10.

If the user is intended to make an accurate prediction, the configuration would serve as a useful tool for an extension. The configuration page is shown in Figure 11. By zooming in and out

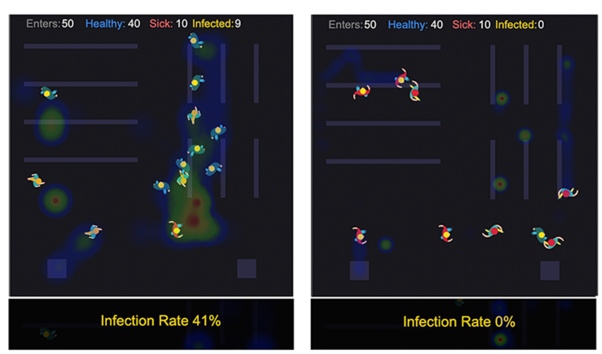

Fig. 9: Comparison of 20 and 2 infected people entered. Left: no measure taken; right: Masks required.

Fig. 10: Shelves in three default crowdedness, number of shelves =15, 20, 25

using the scaling function, drawing boundaries as walls, setting entrance and exit and putting shelves in the original size, users can create their layout similar to the real supermarket.

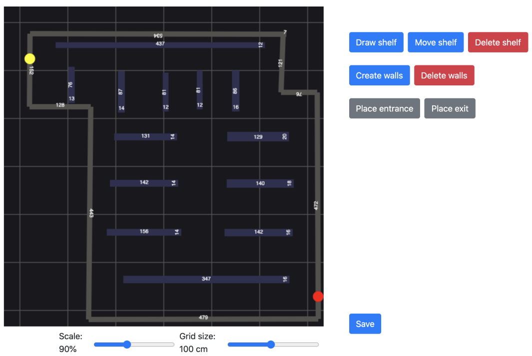

Fig. 11: Configuration page

Which measure should a supermarket owner implement to minimize the possibility of infection?

In our previous use case, we can see that if we set our environment to the default layout, different measures all have an impact on the infection rate. From all the analyses above and our research on the literature, we found that minimizing the amount of people allowed can decrease the infection rate by approximately 10% in a limited duration, while social distance would decrease it by half. Moreover, after forcing people to wear masks, the infection rate significantly drops to near 0%. Therefore, an assumption can be made that the the mask measure is the most effective one, which is also consistent with our literature. We would like to strongly recommend policy makers and owners to suggest all the customers to wear masks while also remind them to keep a social distance. Furthermore, we can also conclude from the time-dimension line chart that having a limitation on the opening time of the supermarket would also be effective as well.

7 Conclusion and Future Work

Overall, the visualization tool of a supermarket simulation fully covers the initial set of questions, allowing users to interact and draw conclusions. On top of that, the website we created is also capable for users to observe virus transmission in real-time. We also believe that it is necessary to reflect on the limitations and results of this visualization model, and to point the way to the possibilities of the future work.

Our work has a lot limitations and yet a lot opportunities to improve. First, our visualization model is more suitable for small and medium-sized supermarkets to use and forecast. A large issue we came across was the simulation of the customers in real time. In the end, we were able to improve the speed of the visualization by implementing the a-star algorithm using web-workers (which is an equivalent to multi-threading). Of course, the performance is in part a problem because of our implementation. We choose JavaScript because of its versatility. However, it is not created for heavy usage and it has to work on all sorts of machines. Therefore it is not very quick. Because of the performance issues, it also took more time than we expected. The hectic schedule contributed little time to make a comparison window, as we imagined beforehand. Also, for a JavaScript page like this would be out of the computation range. Considerably more work will need to be done to show the comparison directly in this page.

Secondly, there are still uncertainties in our calculation model and room for improvement. Although we did some work to show the infection possibility based on the original algorithm and visualized the dangerous area, there is still some progress that can be made in a further project. The path-finding algorithm can still be improved to be more natural and real. Because the algorithm was so computationally heavy, we did not have a lot of resources to implement proper collision avoidance using the algorithm itself. Therefore, we had to implement a more basic collision avoidance function that sometimes creates sub-optimal results.

At last, the most important limitation lies in the fact that we are not able to test the accuracy and precision of our results in a practical and realistic manner. We also realized that there is still some randomness in every trial and it is hard to calculate the mean and standard deviation caused by the limited computational capacity.

From the results in the use case, we can see that the mask measure made the biggest drop in infection rate by 41%, followed by social distance (20%) and limited customers allowed (9%). These results are in line with the literature and news we discussed. Also, half of the opening time for the supermarket would lead to a decrease by 6% approximately. These findings contribute in several ways to our understanding of how the virus spreads in an enclosed public space and provide a basis for policy recommendations worth testing and understanding. There is also abundant room for further progress in demonstrating comparisons and model improvements.

In conclusion, we believe we meet our early requirements and the initial goal of ourselves by successfully producing a usable, simple, visible tool for decision making and cognition of COVID-19 prevention measures. The most obvious finding to emerge from the analysis is that most necessary measure in the supermarket is to limit the customers allowed, and combined with social distance it can also make a difference in the infection rate. In addition, limitation on the supermarket open time can decrease virus transmission as well. Taken together, these results from the visualization suggest that a reasonable measure is to ensure that the forcing customers to wear masks while combined with other measures to prevent the potential risk of infection. The results should however be used with scrutiny.

References

[1] [EB/OL]. https://briangrinstead.com/blog/astar-search-algorithm-in-javascript-updated/.
[2] [EB/OL]. https://www.patrick-wied.at/static/heatmapjs/docs.html.
[3] C. Courtemanche, J. Garuccio, A. Le, J. Pinkston, and A. Yelowitz. Strong social distancing measures in the united states reduced the

covid-19 growth rate. *Health Affairs*, 39(7):1237–1246, 2020. PMID: 32407171.

[4] J. Johnson. Skagit public health. [EB/OL]. https://www.skagitcounty.net/Departments/HealthDiseases/coronavirus.html Accessed April 10, 2021.

[5] L. Peeples. What the data say about wearing face masks. *Nature*, 586:186–189, 2020.

[6] M. Vuorinen, Vuorinenand Aarnio, M. Alava, V. Alopaeus, N. Atanasova, M. Auvinen, and N. Balasubramanian. Modelling aerosol transport and virus exposure with numerical simulations in relation to sars-cov-2 transmission by inhalation indoors. *Safety Science*, 130(130), 2020.