A Framework for Modeling, Analyzing, and Decision-Making in Disease Spread Dynamics and Medicine/Vaccine Distribution

Zenin Easa Panthakkalakath,[1, 2] **Neeraj,** [2] **Jimson Mathew** [2]

[1] Università della Svizzera italiana
[2] Indian Institute of Technology Patna
zenin.easa.panthakkalakath@usi.ch, neeraj.pcs17@iitp.ac.in, jimson@iitp.ac.in

Abstract

The challenges posed by epidemics and pandemics are immense, especially if the causes are novel. This article introduces a versatile open-source simulation framework designed to model intricate dynamics of infectious diseases across diverse population centres. Taking inspiration from historical precedents such as the Spanish flu and COVID-19, and geographical economic theories such as Central place theory, the simulation integrates agent-based modelling to depict the movement and interactions of individuals within different settlement hierarchies. Additionally, the framework provides a tool for decision-makers to assess and strategize optimal distribution plans for limited resources like vaccines or cures as well as to impose mobility restrictions.

Introduction

A century ago, the Spanish flu (Trilla, Trilla, and Daer 2008) infected over 500 million. One-third of the world population at that time - in four successive waves in about two years. Now, we have yet another viral infection that has taken over the world by storm. The outbreak of COVID-19 was first confirmed in Wuhan, China, in December 2019 (Sohrabi et al. 2020). The World Health Organization declared a global public health emergency on 30 January 2020. The pandemic has affected the health of the people and the economy at large, which makes this a huge crisis.

An article in Washington post, titled 'Why outbreaks like coronavirus spread exponentially, and how to "flatten the curve"', written by Harry Stevens (Stevens 2020). The article talks about the mathematical modeling of how diseases spread and the importance of social distancing. The article shows the exponential nature of disease spread using simple animation. It follows an Agent-Based Modelling approach for the same. There have been a few attempts to recreate similar modeling by others in the scientific community. A renowned Mathematician and MATLAB creator, Cleve Moler had created his own version of the COVID-19 simulator in MATLAB and published in his blog (Moler 2020). Similar attempts to reproduce the same using Simulink and Stateflow can be seen in a blog post by Guy Rouleau (Rouleau 2020).

Studies indicate that there has been a massive decrease in the number of people in the central districts of cities due to the pandemic (Lehmann and Alessandretti 2020). It can be interpreted that the reason for such a decline is because there have been a huge number of people who migrated into such centers for economic benefit, now returning to their native region. It is safe to assume that an overwhelming majority of these people come from lower-order settlements like smaller towns and villages around these cities.

We do not know for certain if a vaccine or a cure would be available soon. Even so, we must keep being optimistic and trust the scientific community that we will have our hands on either or both of these soon. In this article, we are assuming that a vaccine would provide immunity to an uninfected person. A cure would enable a sick person to be free of a disease without making the person immune to the disease.

Once such a vaccine or a cure is developed, an administrative decision needs to be taken regarding how the vaccine must be distributed. It is not far-fetched to think that such a vaccine or cure would only be available in limited numbers as the manufacturing facilities would be limited. With the limited amount of resources at hand, the administration bodies must attempt to make optimal decisions to minimize the disease's fatalities. Various parameters would affect such a decision-making process. Some of these are the population density, the number and percentage of infected people, the number and percentage of people who had the disease and recovered, the mobility of the people, quantity of the vaccine or cure available at a given time and the frequency at which the vaccine or cure can be procured.

This article presents a simulation framework that models an environment containing different boundaries and agents moving in the environment. The boundaries depict different settlement hierarchy limits, such as city, town, or village boundaries. The boundaries may or may not be permeable. The agents moving in the environment depict the region's population and have properties to depict whether they are infected, recovered/immune, or uninfected. The framework is parameterized so that the positioning of the boundaries and their impermeability can be changed. Also, the size, location, and the initial number of infected people in a population can be set before the start of the simulation. It is suggested that the choice of modeling method is made based on decision-makers' requirements, type of problem and system complexity, and its characteristics, which could also be derived from economic models (Tako and Robinson 2018, 2012), Borshchev and Filippov (Borshchev and Filippov

2004), Brennan et al. (Brennan, Chick, and Davies 2006), Chahal and Eldabi (Chahal and Eldabi 2010), McHaney et al. (McHaney, Tako, and Robinson 2018).

Preliminaries

Central Place Theory

Introduced in 1933, Central Place Theory is a theory (Berry and Garrison 1958) that attempts to explain the spatial distribution of human settlements. The theory was originally analyzed by a German geographer named Walter Christaller. This theory was the foundation of cities' study as cities' systems, rather than simple hierarchies or single entities. The theory is based on various assumptions. All regions have an isotropic surface with evenly distributed population and resources, having similar cost and quality for goods and services, and consumers having similar purchasing power.

The theory proposes a hexagonal grid (or honeycomb) structure of distribution of cities, towns, and villages. Effectively, there shall be higher-order settlements surrounded by lower-order settlements in the grid. Each higher-order settlements shall be equidistant from other higher-order settlements, and so is the case with lower-order settlements.

Agent-Based Modeling

Agent-Based Modeling is a style of modeling for simulating autonomous agents' actions and interactions in an environment to predict the emergence of complex group behaviors. It is a kind of microscale model that enables simulation of multiple agents performing simultaneous operations and interacting between themselves and the environment in which they reside in. The simulation may also involve different decision-making heuristics and learning rules or adaptive processes. In an analysis, Epstein (Epstein 2009) suggests that ABM is appropriate for modeling pandemics.

The origin of the idea of agent-based modeling can be traced back to the Von Neumann Machine, which was designed in the 1940s. It was a theoretical self-replicating machine in a cellular automata environment. This method has recently been used to model and simulates various scenarios, including, but not limited to, automated driving, cognitive, social simulation, satellite constellation modeling, multi-robot systems, etc.

Framework Development

Our goal is to create a framework that people can use to extend it to model disease spreading scenarios and distribution of vaccines and/or medicine. During the development process, several assumptions were made, and only some features were prioritized.

Assumptions

The following assumptions were made while developing the framework. This was done in order to simplify the development process.

1. All persons in the model are identical. Age, immunity level, or any other factors are not considered. The fatality of a person is randomly determined during the simulation.
2. A person may move freely in accordance with a randomly assigned velocity and direction until it encounters the boundary. The velocity is assigned during the start of the simulation and the magnitude remains unchanged.
3. As a person encounters the boundary, the person may either cross the boundary or get reflected from the boundary. Whether a person bounces off or crosses the boundary is determined in accordance with the probability value assigned to each boundary in the setup.
4. A person may acquire the disease when close to an infected person while within a particular distance from the infected person, and in accordance with the transmission probability.
5. A person who gets cured of the disease without having the medicine shall become immune.
6. An uninfected person who receives the vaccine becomes immune.
7. Only an uninfected person may receive the vaccine.
8. An infected person who receives the medicine becomes equivalent to be uninfected. The person shall not become immune.
9. Only an infected person may receive the medicine.
10. A person who is infected may die in accordance with a probability value.
11. A person may get cured after being infected for a fixed period of time (or iterations).

Features

The key features supported by the framework are:

1. Ability to draw straight line boundaries at different orientations that may be permeable, impermeable, or partial permeable. The user can set a value for the probability of whether a person encountering the boundary would be reflected or transmitted.
2. Ability to define a region inside which people (as agents) can be generated. The center coordinate and the radius of the region need to be specified along with the number of people that need to be generated within that region and the number of people in that who are infected.
3. Ability to specify parameters related to disease spread radius, probability of disease transmission for a person within spread radius of an infected person, probability of getting killed at any time-step while being sick and time period to be free from being sick.
4. Ability to specify parameters related to lockdown implementation. There are parameters to specify at what threshold the ratio of the number of sick people to the total population lockdown must be implemented and revoked. While under lockdown, the people's mobility under lockdown would be multiplied by a lockdown mobility multiplier parameter, which assumes a value between 0 and 1.
5. Ability to distribute vaccine and/or medicines with parameters to specify the quantity, the start time-step, the frequency at which these get replenished and the mode of distribution. The framework supports four modes of

vaccine and medicine distribution based on the statistical state of the agents in the region. These are:

(a) ”equitable”: Distribution proportional to the number of uninfected people in the region at the given time.

(b) ”maximumInfection”: Distribution prioritizing regions with maximum number of infected cases first.

(c) ”maximumUninfected”: Distribution prioritizing regions with maximum number of uninfected people first.

(d) ”infectedAndUninfected”: Distribution prioritizing regions with maximum number of sum of uninfected and infected people first.

Implementation

The framework is implemented in JavaScript with HTML and CSS to support the visualization. It can load all the simulation parameters from a JSON file. These parameters include everything related to boundaries and their orientation, region generation and population information, disease transmission, lockdown implementation, and vaccine and medicine distribution.

The framework loads the boundaries in the orientation specified by the endpoints by reflecting on the boundary parameters specified in the aforementioned JSON file. The line’s color is in accordance with the value of the boundary’s impermeability and the value for the angle of orientation. During the simulation, the agents read the boundary’s color when encountered to compute the impermeability value. If it is decided that the agent is to be reflected from the boundary, the orientation angle comes in handy to calculate the direction of motion post reflection.

People are the agents in this simulation. Uninfected people are represented with a black dot. The color changes to red when they get infect and to green when they recover or get immune. People who die are deleted from the simulation from that point forward.

Simulation Methodology

Using the framework, we can set up different scenarios involving different kinds of borders according to area layouts and people’s simple behavior in different regions. A user of this framework trying to model a city or town of his/her interest can decide on the boundaries and other parameters.

Once the environment model is ready, the user can try simulating without adding any vaccine or cure into the simulation and observe if the data output is similar to what the actual city has been providing in the past many months. If not, the user can keep tweaking the parameters until the values come close to the actual data.

Now that the model is behaving similar to the real-world scenario, we can add in vaccines and/or cure to the equation and run different simulations and observe the results. Try to find the decisions involving the least amount of fatality, which is the decision that the administration must make regarding vaccine distribution.

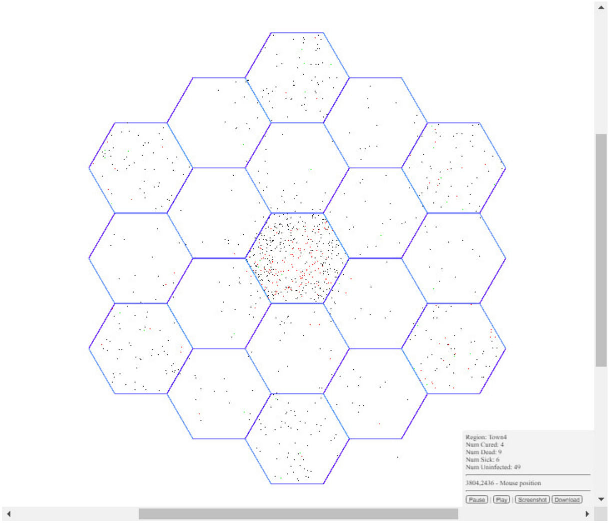
Below mentioned are the steps that one must follow to use the framework to model for their region.

1. Create a new JSON file named ’myRegion.json’. Edit the variable named ’environmentJSONFile’ in ’script.js’ file to point to this file.
2. Start by drawing the boundaries. Create a variable named ”boundaries”, which is an array of arrays. Each child array represents a boundary, with each of its elements being X-coordinate of the first point, Y-coordinate of the first point, X-coordinate of the second point, Y-coordinate of the second point, and impermeability. The value of impermeability ranges between 0 and 1, 0 means completely permeable, and 1 means completely impermeable.
3. Define the region inside which persons get generated. Add in a variable named ”regions,” which is an array of objects. Each object corresponds to a circular region inside which the specified number of people gets generated. The properties of the object are ”id”, ”population”, ”infected”, ”center”, ”radius” and ”mobilityFactor”, which corresponds to a string identity of the region, total population in the region, number of infected people in the region, geographic center coordinate of the region (an array having two values corresponding to X-coordinate and Y-coordinate), the radius of the region and mobility factor (which corresponds to the velocity of the people in the simulation).
4. Finally, define other environmental parameters ”boundaryThickness”, ”spreadRadius”, ”curePeriod” and ”killProbability”. Environmental parameters like ”lockdownStartThreshold”, ”lockdownEndThreshold” and ”lockdownMobilityMultiplier”. These may be chosen in accordance with how the Government usually reacts to the situation. Environmental parameters related to vaccine and medicine distribution may be set to a value such that it wouldn’t make any impact, say we set the quantity to zero.
5. Run the simulation, download the result, and observe whether the simulation results are coherent with the disease spread information for the city at hand. If not, check and modify the environmental parameters until the data is coherent enough.
6. Now, modify the parameters related to vaccine and medicine distribution in accordance with the potential future availability of the same.

To showcase the framework’s capability, we have come up with a region layout involving hexagonal grids. The hexagons in the grid would represent villages, towns, or cities. The Central Place Theory inspires this layout and the modeling of many cities may be similar to this.

The hexagonal grids were programmatically generated and the values for boundary impermeability and the values for initial population and infected people in each region were entered manually. As explained in the aforementioned points, the parameters were tweaked until a reasonable simulation result was observed with the values for vaccine and medicine quantities set to zero.

In all the scenarios that we are trying to simulate, we are keeping some parameters constant. The values for these parameters can be seen in Table 1.



Region: Town4
Num Cured: 4
Num Dead: 9
Num Sick: 6
Num Uninfected: 49

3804,2436 - Mouse position

Figure 1: A screenshot showcasing the hexagonal grid (honeycomb) layout depicting different settlement hierarchies and dots depicting the population in the region.

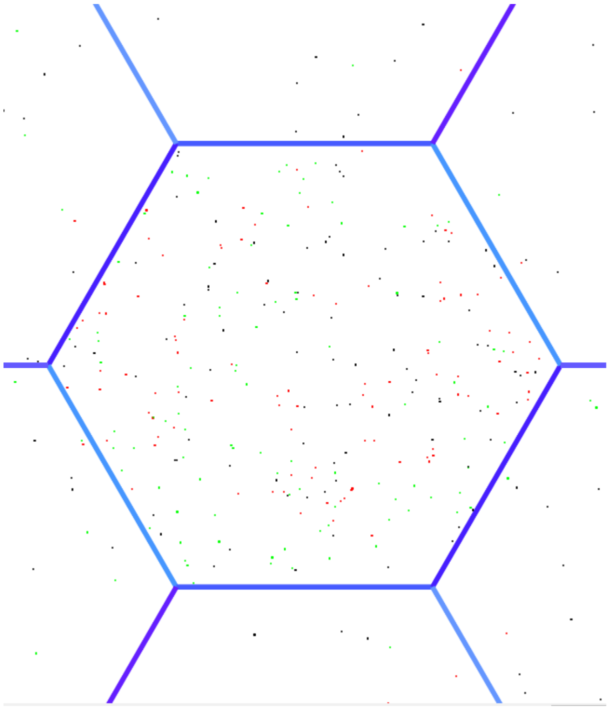


Figure 2: A screenshot focusing the central city location. Here we can clearly see the dots with different colors. Black depicts people who are uninfected, red depicts people who are currently infected, and green depicts people who have recovered from the disease and are immune to it.

In the hexagonal grid, the city, towns, and villages were placed, as shown in Figure 3. The total population, the number of infected people in each region type, and the people's mobility factor in these regions are mentioned in Table 2.

Parameter name	Value
spreadRadius	5
curePeriod	250
killProbability	0.005
transmissionProbability	0.7

Table 1: Parameters used in the simulation that were kept constant across various scenarios.

Hierarchy	Population	Infected	Mobility
City	750	5	1
Town	100	5	3
Village	15	1	5

Table 2: Initial conditions for different settlement hierarchies assumed in the simulation. The values may be considered scaled.

The value of the impermeability of the borders were decided based on who shares the border. Even during a pandemic, the border between a city and a village would be more permeable compared to the the border between a village and a town, which would be more permeable that the border between two villages. This is because people from villages would be dependent on nearby towns and cities for economic activities. Even during a pandemic situation, such an activity would be required to ensure survival of the population. The values of the same that we adopted for this simulation is shown in Table 3.

Between hierarchies	Impermeability
City-Village	0.7
Town-Village	0.8
Village-Village	1

Table 3: The border impermeability values between different settlement hierarchy.

Arranging cities, villages, and towns in such an arrangement enables capturing the phenomenon of population migration from higher orders of settlements to lower orders of settlements during a pandemic time. It is also important to differentiate the central city's statistical data to that of the entire region. A correlation between these two sets of statistics is important, as a survey that doesn't consider such a migration would have missing links.

Results

After the development, the framework was put to test to observe the simulation results under various circumstances. These are described in the following subsections. Note that the simulation was conducted for a hypothetical hexagonally divided regions having population scaled to smaller numbers.

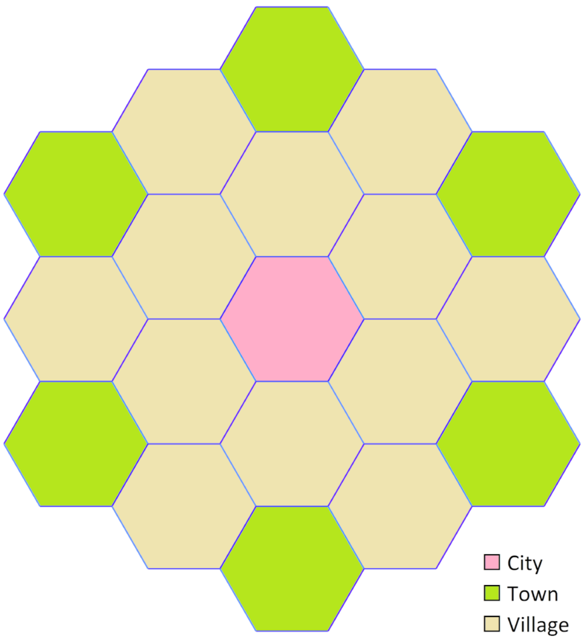


Figure 3: An image depicting the placement of city, towns and villages in a hexagonal grid.

Simulation Results without Vaccine, Cure or Lockdown

If no restrictions are imposed by the administrative authorities, the scenario would end up something similar to what is depicted in Figures 4 and 5.

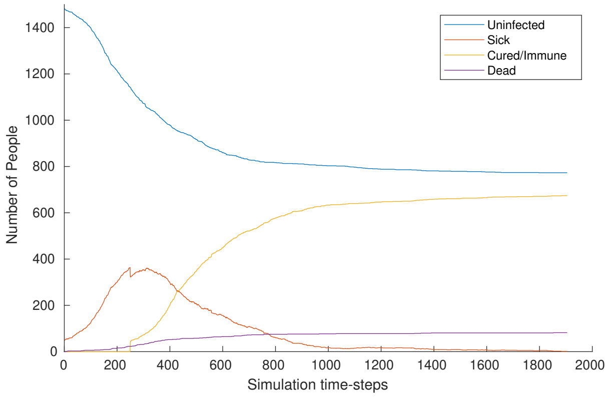


Figure 4: The aggregate simulation results for all the regions when no vaccine, medicine or lockdown were implemented.

At the end of the simulation, the statistics of the number of people who where dead and recovered are mentioned in Table 4.

Simulation Results with lockdown

Simulation was conducted wherein lockdown was implemented whenever more than 10% were infected and revoked whenever less than 2% were infected in any particular region. The parameter values for the same is described in Table 5.

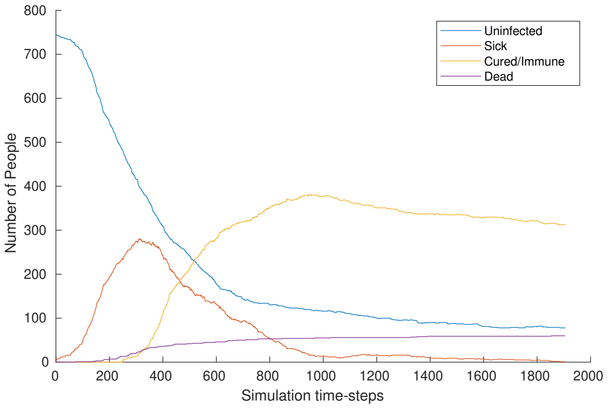


Figure 5: The simulation results for the central city when no vaccine, medicine or lockdown were implemented.

Parameter name	Value
Initial Population	1530
Simulation Period	1905
Total Immune	675
Total Dead	82

Table 4: Tabulated final results for all the regions when no vaccine, medicine or lockdown were implemented.

Parameter name	Value
lockdownStartThreshold	0.1
lockdownEndThreshold	0.02
lockdownMobilityMultiplier	0.1

Table 5: Simulation parameters used to implement lockdown.

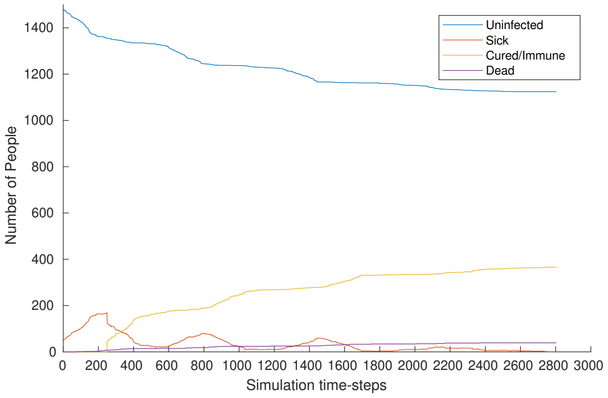


Figure 6: The aggregate simulation results for all the regions when lockdown was implemented.

At the end of the simulation, the statistics of the number

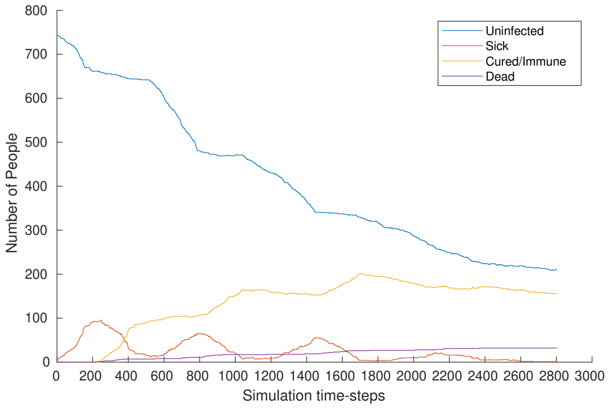


Figure 7: The simulation results for the central city when lockdown was implemented.

of people who where dead and recovered are mentioned in Table 6.

Name	Value
Initial Population	1530
Simulation Period	2804
Total Immune	367
Total Dead	39

Table 6: Tabulated final results for all the regions with lockdown.

We can observe that there would be four waves when a lockdown is implemented in such a manner.

Simulation Results with Vaccine distribution

Simulations were conducted with all four supported methods of vaccine distribution. These are mentioned in the subsequent paragraphs, figures and tables. In all the cases, vaccine distribution starts at 300 time-step with a quantity of 200 units and the quantity gets replenished after every 100 subsequent time-step.

Firstly, let us observe the results when vaccine is distributed equitably i.e., proportional to the number of uninfected people in the region at the given time. The parameter values for the same is described in Table 7.

Parameter name	Value
vaccineDistributionStartTime	300
vaccineDistributionFrequency	100
vaccineDistributionQuantity	200
vaccineDistributionMechanism	”equitable”

Table 7: Simulation parameters used when vaccine distribution mode was ”equitable”.

At the end of the simulation, the statistics of the number of people who where dead and recovered are mentioned in Table 8.

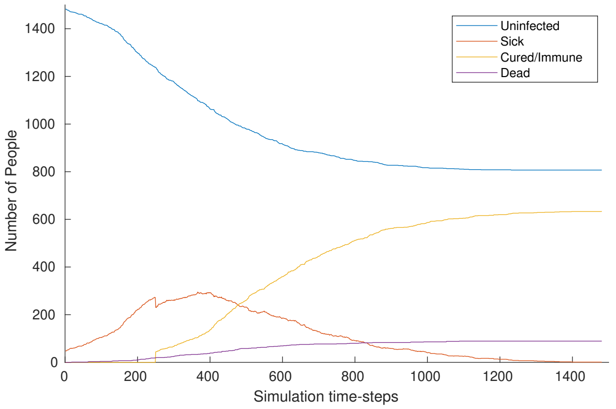


Figure 8: The aggregate simulation results for all the regions when vaccine distribution mode was ”equitable”.

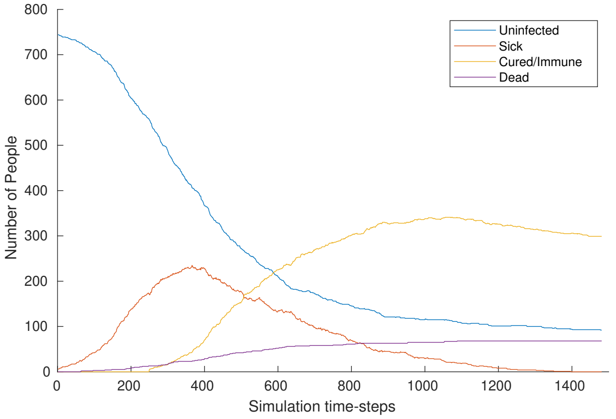


Figure 9: The simulation results for the central city when vaccine distribution mode was ”equitable”.

Name	Value
Initial Population	1530
Simulation Period	1481
Total Immune	634
Total Dead	89

Table 8: Tabulated final results for all the regions with vaccine distribution mode ”equitable”.

Secondly, let us observe the results when vaccine is distributed prioritizing regions with maximum number of infected cases first. The parameter values for the same is described in Table 9.

Parameter name	Value
vaccineDistributionStartTime	300
vaccineDistributionFrequency	100
vaccineDistributionQuantity	200
vaccineDistributionMechanism	”maximumInfection”

Table 9: Simulation parameters used when vaccine distribution mode was ”maximumInfection”.

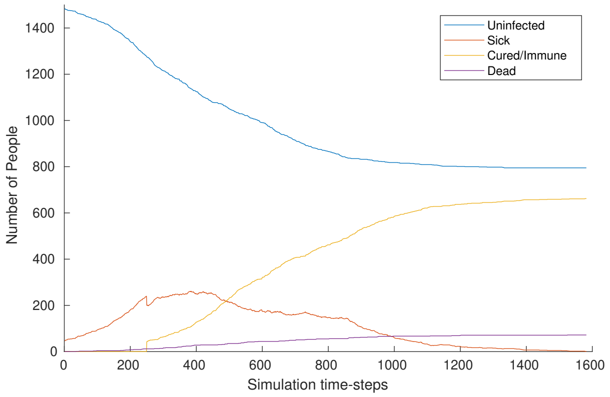


Figure 10: The simulation results for all the regions when vaccine distribution mode was ”maximumInfection”.

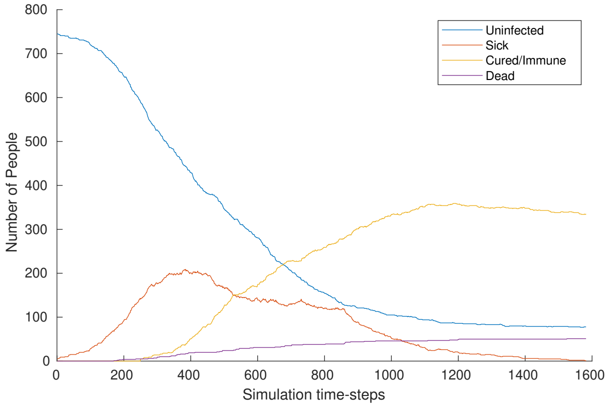


Figure 11: The simulation results for the central city when vaccine distribution mode was ”maximumInfection”.

At the end of the simulation, the statistics of the number of people who where dead and recovered are mentioned in Table 10.

Name	Value
Initial Population	1530
Simulation Period	1481
Total Immune	663
Total Dead	72

Table 10: Tabulated final results for all the regions with vaccine distribution mode ”maximumInfection”.

Thirdly, let us observe the results when vaccine is distributed prioritizing regions with maximum number of uninfected people first. The parameter values for the same is described in Table 11.

At the end of the simulation, the statistics of the number of people who where dead and recovered are mentioned in Table 12.

Fourthly, let us observe the results when vaccine is distributed prioritizing regions with highest sum of infected and uninfected people first. The parameter values for the same is described in Table 13.

At the end of the simulation, the statistics of the number of people who where dead and recovered are mentioned in Table 14.

Parameter name	Value
vaccineDistributionStartTime	300
vaccineDistributionFrequency	100
vaccineDistributionQuantity	200
vaccineDistributionMechanism	”maximumUninfected”

Table 11: Simulation parameters used when vaccine distribution mode was ”maximumUninfected”.

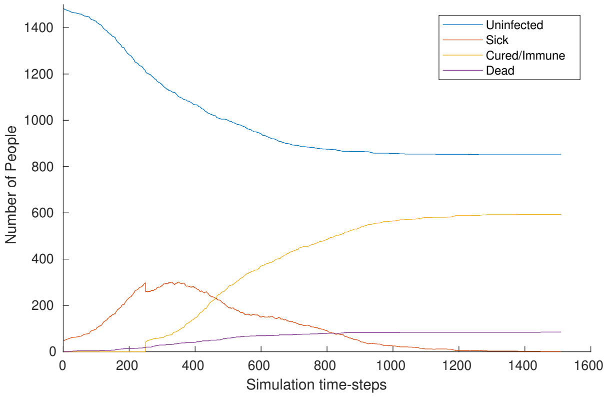


Figure 12: The simulation results for all the regions when vaccine distribution mode was ”maximumUninfected”.

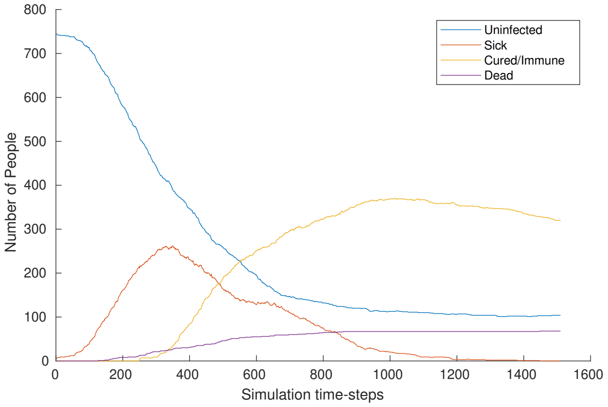


Figure 13: The simulation results for the central city when vaccine distribution mode was ”maximumUninfected”.

Name	Value
Initial Population	1530
Simulation Period	1510
Total Immune	594
Total Dead	85

Table 12: Tabulated final results for all the regions with vaccine distribution mode ”maximumUninfected”.

Parameter name	Value
vaccineDistributionStartTime	300
vaccineDistributionFrequency	100
vaccineDistributionQuantity	200
vaccineDistributionMechanism	”infectedAndUninfected”

Table 13: Simulation parameters used when vaccine distribution mode was ”infectedAndUninfected”.

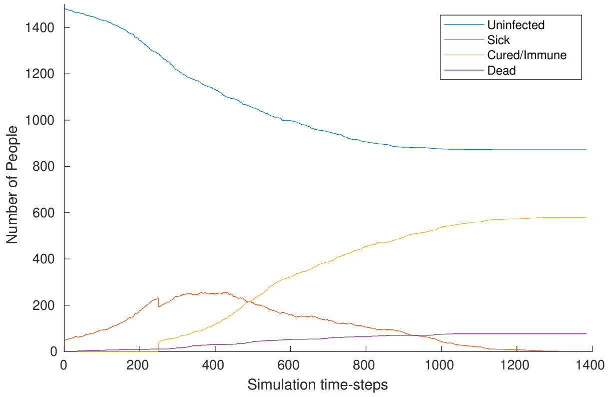


Figure 14: The simulation results for all the regions when vaccine distribution mode was ”infectedAndUninfected”.

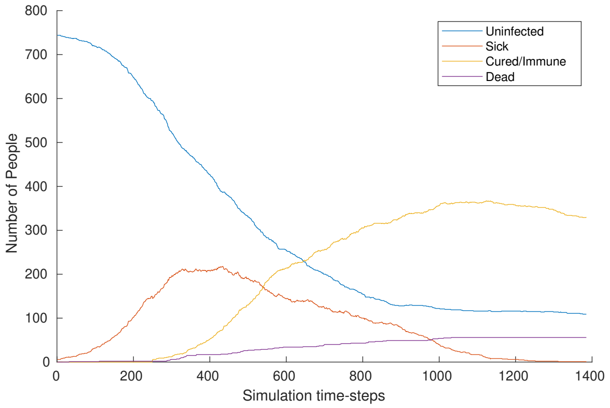


Figure 15: The simulation results for the central city when vaccine distribution mode was ”infectedAndUninfected”.

Name	Value
Initial Population	1530
Simulation Period	1385
Total Immune	581
Total Dead	77

Table 14: Tabulated final results for all the regions with vaccine distribution mode ”infectedAndUninfected”.

Medicine only distribution

Firstly, let us observe the results when medicine is distributed equitably i.e., proportional to the number of uninfected people in the region at the given time. The parameter values for the same is described in Table 15.

Parameter name	Value
medicineDistributionStartTime	300
medicineDistributionFrequency	100
medicineDistributionQuantity	200
medicineDistributionMechanism	”equitable”

Table 15: Simulation parameters used when medicine distribution mode was ”equitable”.

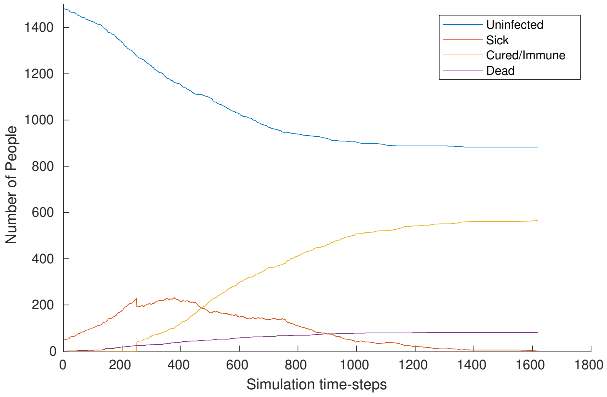


Figure 16: The aggregate simulation results for all the regions when medicine distribution mode was ”equitable”.

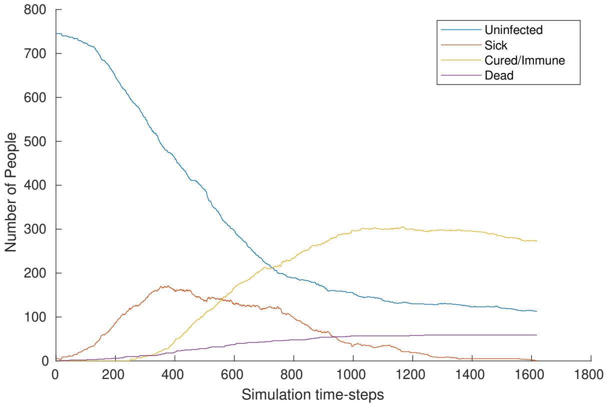


Figure 17: The simulation results for the central city when medicine distribution mode was ”equitable”.

At the end of the simulation, the statistics of the number of people who where dead and recovered are mentioned in Table 16.

Secondly, let us observe the results when medicine is distributed prioritizing regions with maximum number of infected cases first. The parameter values for the same is described in Table 17.

At the end of the simulation, the statistics of the number of people who where dead and recovered are mentioned in Table 18.

Name	Value
Initial Population	1530
Simulation Period	1618
Total Immune	566
Total Dead	81

Table 16: Tabulated final results for all the regions with medicine distribution mode ”equitable”.

Parameter name	Value
medicineDistributionStartTime	300
medicineDistributionFrequency	100
medicineDistributionQuantity	200
medicineDistributionMechanism	”maximumInfection”

Table 17: Simulation parameters used when medicine distribution mode was ”maximumInfection”.

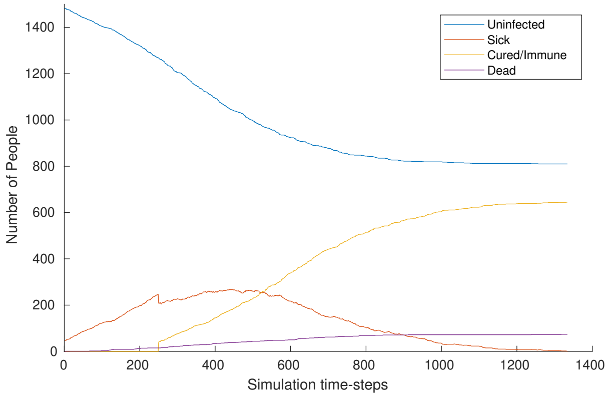


Figure 18: The simulation results for all the regions when medicine distribution mode was ”maximumInfection”.

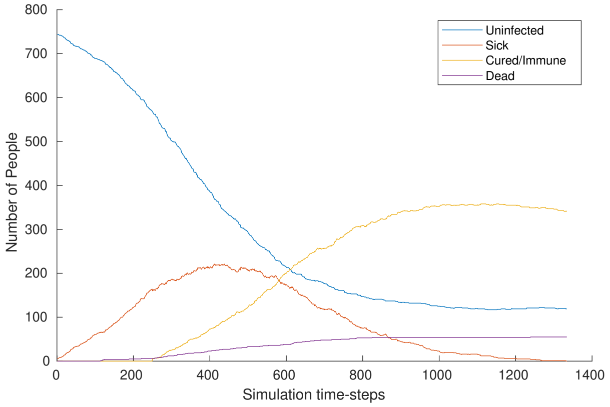


Figure 19: The simulation results for the central city when medicine distribution mode was ”maximumInfection”.

Thirdly, let us observe the results when medicine is distributed prioritizing regions with maximum number of uninfected people first. The parameter values for the same is described in Table 19.

Name	Value
Initial Population	1530
Simulation Period	1334
Total Immune	645
Total Dead	75

Table 18: Tabulated final results for all the regions with medicine distribution mode ”maximumInfection”.

Parameter name	Value
medicineDistributionStartTime	300
medicineDistributionFrequency	100
medicineDistributionQuantity	200
medicineDistributionMechanism	”maximumUninfected”

Table 19: Simulation parameters used when medicine distribution mode was ”maximumUninfected”.

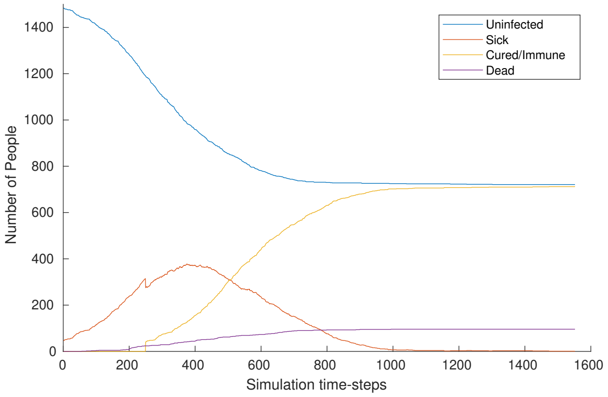


Figure 20: The simulation results for all the regions when medicine distribution mode was ”maximumUninfected”.

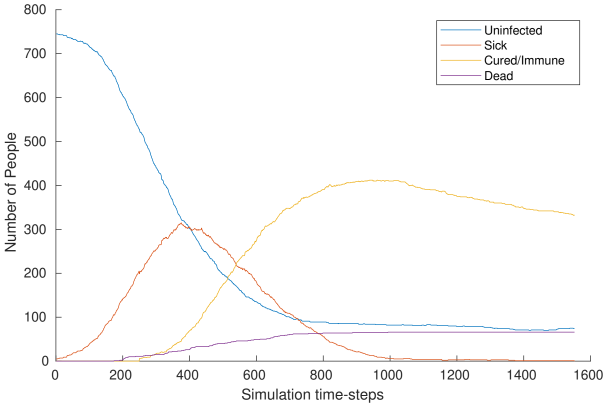


Figure 21: The simulation results for the central city when medicine distribution mode was ”maximumUninfected”.

At the end of the simulation, the statistics of the number of people who where dead and recovered are mentioned in Table 20.

Fourthly, let us observe the results when medicine is distributed prioritizing regions with highest sum of infected and uninfected people first. The parameter values for the same is

Name	Value
Initial Population	1530
Simulation Period	1553
Total Immune	713
Total Dead	96

Table 20: Tabulated final results for all the regions with medicine distribution mode ”maximumUninfected”.

described in Table 21.

Parameter name	Value
medicineDistributionStartTime	300
medicineDistributionFrequency	100
medicineDistributionQuantity	200
medicineDistributionMechanism	”infectedAndUninfected”

Table 21: Simulation parameters used when medicine distribution mode was ”infectedAndUninfected”.

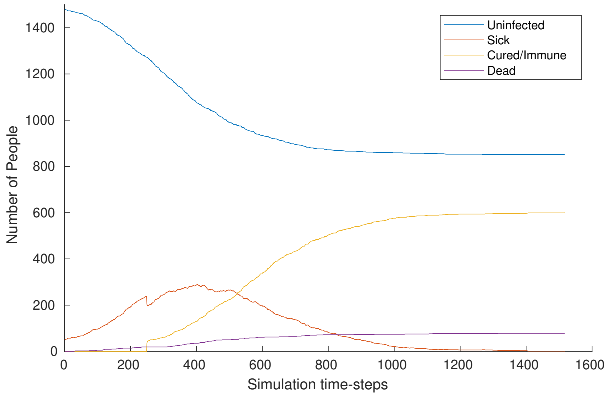


Figure 22: The simulation results for all the regions when medicine distribution mode was ”infectedAndUninfected”.

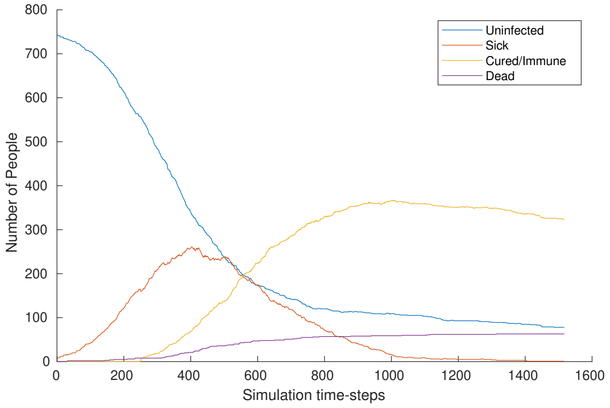


Figure 23: The simulation results for the central city when medicine distribution mode was ”infectedAndUninfected”.

At the end of the simulation, the statistics of the number of people who where dead and recovered are mentioned in Table 22.

Name	Value
Initial Population	1530
Simulation Period	1517
Total Immune	600
Total Dead	78

Table 22: Tabulated final results for all the regions with medicine distribution mode ”infectedAndUninfected”.

Simultaneous Vaccine and Medicine distribution

With medicine and vaccine distribution starting at 300 timestep with a quantity of 200 units and getting replenished after every 100 time-steps -Vaccine prioritizing regions with maximum number of the sum of uninfected and infected people first and medicine prioritizing regions with maximum number of infected people first - Lockdown enforced when more than 10% infected and revoked when less than 2% infected. The parameter values for the same is described in Table 23.

Parameter name	Value
lockdownStartThreshold	0.1
lockdownEndThreshold	0.02
lockdownMobilityMultiplier	0.1
vaccineDistributionStartTime	300
vaccineDistributionFrequency	100
vaccineDistributionQuantity	200
vaccineDistributionMechanism	”infectedAndUninfected”
medicineDistributionStartTime	300
medicineDistributionFrequency	100
medicineDistributionQuantity	200
medicineDistributionMechanism	”maximumInfection”

Table 23: Simulation parameters used when medicine distribution mode was ”infectedAndUninfected”.

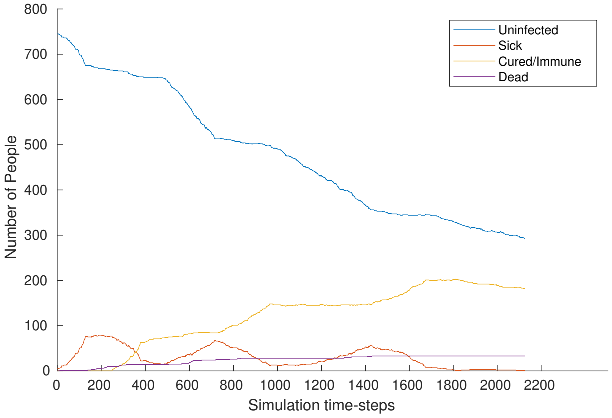


Figure 25: The simulation results for the central city with lockdown imposition, and, vaccine and medicine distribution

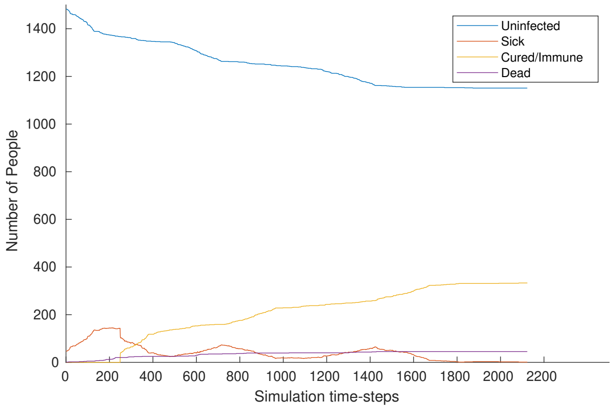


Figure 24: The simulation results for all the regions with lockdown imposition, and, vaccine and medicine distribution

At the end of the simulation, the statistics of the number of people who where dead and recovered are mentioned in Table 24.

Name	**Value**
Initial Population	1530
Simulation Period	2123
Total Immune	334
Total Dead	45

Table 24: Tabulated final results for all the regions with medicine distribution mode ”infectedAndUninfected”.

Conclusions

The designed framework is able to simulate wide variety of scenarios. Indeed, there are infinitely many possible combinations of vaccine and medicine distribution that can be, and some of these were explored in this article. Based on different social and economic conditions, the decision makers may tweak different parameters to virtually test their decision making skills in the future using this framework. However, do note that the framework does have some assumptions, and it may be improved in the future to have less of them.

References

Berry, B. J.; and Garrison, W. L. 1958. Recent developments of central place theory. *Papers in Regional Science*, 4(1): 107–120.

Borshchev, A.; and Filippov, A. 2004. From system dynamics and discrete event to practical agent based modeling: reasons, techniques, tools. In *Proceedings of the 22nd international conference of the system dynamics society*, volume 22, 25–29. Oxfort England.

Brennan, A.; Chick, S. E.; and Davies, R. 2006. A taxonomy of model structures for economic evaluation of health technologies. *Health economics*, 15(12): 1295–1310.

Chahal, K.; and Eldabi, T. 2010. A multi-perspective comparison for selection between system dynamics and discrete event simulation. *International Journal of Business Information Systems*, 6(1): 4–17.

Epstein, J. M. 2009. Modelling to contain pandemics. *Nature*, 460(7256): 687–687.

Lehmann, S.; and Alessandretti, L. 2020. Population redistributed!

McHaney, R.; Tako, A.; and Robinson, S. 2018. Using LIWC to choose simulation approaches: A feasibility study. *Decision Support Systems*, 111: 1–12.

Moler, C. 2020. COVID-19 Simulator.

Rouleau, G. 2020. COVID-19: Simulating exponential spread in Simulink.

Sohrabi, C.; Alsafi, Z.; O’neill, N.; Khan, M.; Kerwan, A.; Al-Jabir, A.; Iosifidis, C.; and Agha, R. 2020. World Health Organization declares global emergency: A review of the 2019 novel coronavirus (COVID-19). *International journal of surgery*, 76: 71–76.

Stevens, H. 2020. Why outbreaks like coronavirus spread exponentially, and how to “flatten the curve”.

Tako, A. A.; and Robinson, S. 2012. The application of discrete event simulation and system dynamics in the logistics and supply chain context. *Decision support systems*, 52(4): 802–815.

Tako, A. A.; and Robinson, S. 2018. Comparing discrete-event simulation and system dynamics: Users’ perceptions. *System dynamics: Soft and hard operational research*, 261–299.

Trilla, A.; Trilla, G.; and Daer, C. 2008. The 1918 “spanish flu” in spain. *Clinical infectious diseases*, 47(5): 668–673.

Appendix

Code and data

The implementation code, generated data, results and analysis are available in https://github.com/zenineasa/frameworkForVaccineAndCureDistribution.

Hardware and software setup

The specifications for the device and the software libraries used to perform the simulations are described in Table 25 and Table 26. However, note that the framework developed to run on any modern computer using a modern browser, irrespective of the operating system. Although we used MATLAB for analysing and interpreting the results, one could alternatively use Python or Excel.

Component	**Specification**
Hardware	Lenovo ideapad 330-15ARR
Processor	AMD Ryzen 5 2500U
No. of cores	8
Graphics	AMD Radeon vega 8 graphics
Memory	8 GB (2× 4 GB DDR4 @ 2400 MHz)
OS	Ubuntu 20.04.02 LTS

Table 25: Specification of the machine used for performance evaluation

Software / Library	**Version**
Google Chrome	86.0.4240.183
MATLAB	R2020b

Table 26: Versions of different software and libraries used

Choosing to build a new framework

The reason for building a framework that can run on a modern browser was motivated by the fact that these are free to use, have immense capabilities, and are nearly universally accessible. Decision makers need not require programmers to try different strategies and observe the potential outcome of their decisions. Ease of use and access were our primary priority while developing this framework.