

Contact tracing effectiveness evaluation in epidemic scenarios using agent based models

Students: Naomi Bazlov, Honi Pein, Laetitia Codd
Mentors: Charlotte Zemmell, Pau Batlle, Joel Rehmann
Subject: Computer Science and Philosophy

Abstract

In this work we explored how agent-based modelling techniques can be used to predict the development of a pandemic in a population. In our project, we utilised Agent Based Models (ABMs), computer models that attempt to capture the behaviour of individuals within an environment¹. We explored how we can use Python to create an agent-based model of a pandemic, specifically the movement of a population and its relationship with the spread of the virus, and the philosophical justifications for doing so. We found that we are able to gain some insight into how a pandemic evolves, and defend our model adequately by considering values in science. Our model shows that the effects of contact tracing only become pronounced when tracing is moderately honoured by the majority of the population, or strictly honoured by some of the population.

1 Introduction

Our aim was to investigate the effect of contact tracing in preventing the spread of an infectious disease via simulating the pandemic in a model, by varying the proportion of the population that is being traced. Human behaviour is a crucial factor both in the spread of the disease and in controlling it, therefore, modelling human behaviour is an important tool in improving the control effort.

Modelling as a device is used in all fields of science, from the ‘standard model’ of particle physics to mathematical modelling of complex biological systems. In fact, it is hard to find an area of modern science that does not engage in the activity of modelling. In particular, modelling and other computational devices have been invaluable in advancing quantitative science such as Evidence Based Medicine. There has been an especially large increase in interest in agent-based modelling, the type of modeling that we decided to use for our project, largely due to its ability to make significant projections in the absence of data.

Agent Based Models are models that attempt to capture the behaviour of individuals within an environment. They are often more intuitive than other types of models, as they represent objects as we see them: individual agents in the world. ABMs are often used to capture what we understand

¹<http://www.geog.leeds.ac.uk/courses/other/crime/abm/general-modelling/index.html>, accessed on 21 July 2020

about certain systems, and test that understanding; comparing the results of predictive models to empirical data shows us how accurate the theory that informed the model is.

In examples such as pandemics, however, these models can also be used to make predictions in the absence of sufficient evidence. When a disease is the sufficient combination of infectious and deadly to warrant acting quickly over acting precisely, ABMs can be used to make quick and immediate predictions based on theory that can help inform emergency policy decisions. Philosopher of medicine Dr Jonathan Fuller has recently ² argued that modelling has been a particularly important device in the coronavirus pandemic because the data we have been able to garner has been limited and very low quality. To illustrate this point, Professor John P.A. Ioannidis labelled the early months of 2020 an “evidence fiasco” ³. Thus, theoretical methods, like models, were more practically useful than evidence-based ones. This de-emphasis on first hand evidence has a large impact on the kinds of inferences we are justified in drawing from ABMs, a topic we explore in detail below.

However, the fact that we have to rely on models to offer prognoses of this pandemic is an imperfect scenario, and many agree that we should not be complacent in collecting evidence. There are also, of course, drawbacks to using ABMs, which should be acknowledged. Specifically, deciding how many details to include in the model is a complex task. A large amount of parameters, features, and behaviors will render the programming process much more difficult, but a smaller amount will reduce the nuance and accuracy of the model. A more practical drawback is the steep learning curve involved in learning to build an ABM, which is something we have had to overcome in this project.

The simple SIR (Susceptible-Infected-Recovered) model has been a basis for the majority of disease models studied since 1927, as pointed out in the 2010 review article by Funk, Salathé and Jansen⁴. In this model, the individuals in the population transition from S to I to R states at a certain rate; more sophisticated models include spatial or contact network structure, so that an individual can only be infected by a constrained set of other individuals. An example of an SIR model that has been extremely influential in offering an estimation of the pandemic’s projection is Professor Neil Ferguson and his team in Imperial College London’s simulation of the impact of social distancing measures ⁵.

One difficulty in the SIR-type models is quantifying the effect of a behavioural change on the rates of transition between the states; ideally this should be inferred from the empirical data, however this is impeded by the low quality and limited availability of COVID-19 data on the impact of social distancing measures. In light of this, in our project we attempt to model this effect stochastically. In our model, the infection spread is conditioned by spatial proximity of individuals, who move about randomly. We can thus model a change in behaviour by controlling the parameters of the affected individual’s movement; the effect on the S-I-R transition rate is brought about by changes in the individual’s trajectory and does not need to be guessed. While some of the parameters of that behavioural change do have to be guessed, it is still true that by this method qualitative changes can be understood more quantitatively. We believe that our approach is more natural in modelling

²<http://bostonreview.net/science-nature/jonathan-fuller-models-v-evidence>, accessed on 22 July 2020

³<https://www.statnews.com/2020/03/17/a-fiasco-in-the-making-as-the-coronavirus-pandemic-takes-hold-we-are-making-decisions-without-reliable-data/>, accessed on 25 Jul 2020

⁴<https://royalsocietypublishing.org/doi/10.1098/rsif.2010.0142>, accessed on 24 Jul 2020

⁵<https://www.imperial.ac.uk/mrc-global-infectious-disease-analysis/covid-19/report-12-global-impact-covid-19/>, accessed on 25 July 2020

behavioural changes, and should be relevant for studies that aim to inform centralised policy in controlling the spread of a disease.

It should be noted that our stochastic modelling approach is significantly more resource-heavy than approaches to date, based on ordinary differential equations with or without fixed contact network constraints. This type of modelling requires recent hardware, however the simulations are still time-consuming. Adopting our method on a large scale would involve the use of frameworks designed for parallel computation and/or the usage of more advanced simulation algorithms.

2 Theoretical model

Initially we started out with an SIR model, but later decided to add complexity by adding a presymptomatic and asymptomatic stage for our agents, which made our model closer to an SEIR (Susceptible-Exposed-Infected-Recovered) model. We thought this was a better representation of the real life situation; however, the “exposed” agents were counted in the same category as infected in our graphs for simplicity.

In our Agent Based Model, the individual agents represent humans and are modelled as particles moving on a board, or map. The simulation is split into frames of time rather than being continuous. This significantly reduces the volume of information that needs to be stored which is a great benefit as we can complete simulations faster. Another way to save on data storage is by making the model Markovian (the future of the model is independent of the past, so previous states are discarded).

Movement is modelled in the following way: each agent chooses a point on the board, and moves towards it during the next several frames of the simulation; when they reach their destination, they have a certain probability of either choosing a new point to move towards or remaining in the same spot. We chose this way to model the agents’ movement as we thought it was a good simplified representation of how humans move in real life, and the random nature of the walking removes the dependency of the results we obtained on movement patterns. Crucially, we can assume that the observed change in the spread of disease when we vary other parameters is independent of the way agents walk, giving greater justification to the causal relationships we inferred from the model, specifically between track and trace and the rate of spread of the virus.

One of the most important parameters in the model is the contagion radius. When a susceptible agent approaches within the contagion radius of an infected agent, it has a set probability of getting infected from that person. The agents have an initial incubation phase once they get infected; this models the real life occurrence of presymptomatic people and there is a probability of transitioning into the visibly symptomatic stage after each frame. This was crucial for the tracing part of the model, as presymptomatic agents were not traced.

To represent tracking and tracing in our model, we first chose which agents would be being traced. At the start of the simulation, each agent had a certain probability of being included in the tracing and we varied this to investigate how widespread tracing must be to have an effect on the overall situation. If a pair of agents who were being traced stood within a certain range of each other for several frames in a row, and one later became visibly infected, the other would be notified to take appropriate isolation measures. These measures were modelled by a reduction in the agent’s radius of contagion and speed, and these were kept constant for all agents within each simulation.

We considered many other parameters, for instance, the size of the board and the population density. After running many simulations on differently sized boards with the same population density we found that the size of the board did not cause any changes outside of the natural error margin of the model, which showed that our results are independent of the size of the board used. Of course, in bigger boards the number of agents was larger and this improved the quality of the results by reducing the effects of anomalies. The drawback to using a large board is the drastic increase of computational power and time needed to run the entire simulation, so a compromise was necessary.

3 Implementation

We built our model on Python3. In order to plot graphs and animate the simulation itself, we used the matplotlib module; we used numpy to define all the functions that gave random outcomes, and pandas to save numerical data easily in compressed file formats.

Our code is written in the style of Object Oriented Programming (OOP). In OOP, objects with similar properties or attributes are defined as instances of a “class” or general blueprint. The class also contains methods, which are essentially the rules according to which the objects behave and exchange data. In our simulation, we created three classes: Walker, Board and Model. Objects of the Walker class are walkers, or humans; objects of the Board class are the boards or maps upon which the walkers travel, the Model class is used to store variables of the pandemic. Here are some parameters that we kept constant:

- Infect probability: chance that an infected agent gives the virus to a susceptible agent within his radius. This was 0.5 throughout
- Initial infect radius: this is the radius that presymptomatic and infected walkers have to infect others, and was always initially 5 arbitrary units of the board. The radius was reduced when app users were alerted to isolate.
- Presymptomatics: When a walker is first successfully infected, he becomes presymptomatic with probability 0.6 and asymptomatic with probability 0.4. Asymptomatic walkers have a constant infection radius of 2 board units (to represent the fact that they are much less infectious than presymptomatic walkers)
- State transition probabilities: At each frame of the model, walkers who are presymptomatic have a set probability of becoming visibly infected, which is 0.01. Infected walkers have a 0.01 probability of becoming recovered/dead.

In our executable code, we first defined a board, then used an `add_walker` method to initialise objects of the class Walker that “belonged” to this board:

```
myboard = Board(100,100)
myboard.add_walker(status=susceptible)
```

The `add_walker` method was looped so that the number of walkers on the board was equal to the population density multiplied by the board area.

The board was used as a storage for the data of all the walkers, since each individual walker does not have any information about the attributes of any other walkers (their location, health status etc.) This information was used in the very important `find_infected` and `find_app_users` methods, which created a list of infected or app-using walkers respectively.

The board updated each frame until there were no infectious walkers left on the board (after which no changes in health status could occur in the model). The code looked a bit like this:

```
define the update function:
    for walker in the board:
        walker.walk()
        walker.update_status()
```

The `walk` method moved the walker a step closer to its randomly chosen destination, and `update_status` checked whether the walker's health status changed - for susceptible agents, this involved using the board's `find_infected` method to find infected agents in the vicinity, and for others this involved rolling a weighted die to find out if they transitioned to the next state.

We used `matplotlib` to create an animated simulation of the board; this was not instrumental to collecting data but was extremely useful to visualise the situation. The differently coloured dots moved around on the plotted board. At each frame, we recorded the number of susceptible, infected and recovered walkers in three lists and plotted an SIR graph with the data at the end of the simulation (no. of walkers on the y axis against number of frame on the x axis) - again, this was done with `matplotlib`.

Using `pandas`, we exported the final number of iterations and susceptible, infected and recovered people into a csv file. This allowed us to later plot the proportion of walkers who remained susceptible (hadn't caught the virus at all) against the proportion of walkers that were being traced during the simulation to conclusively show that implementing contact tracing can effectively reduce disease transmission.

4 Results

In our first experiment, we made the walkers' speed when alerted slow down to 0, and varied the percentage of walkers being traced and by what factor their radius of infection was reduced. The total number of walkers in each simulation was 542 (due to the rounding of the number of susceptible and infected walkers in the initial board setup). The results can be observed in Figure 1.

In the case that alerted walkers' radius became 0, 39 susceptible walkers out of 542 remained when no walker was traced as opposed to 451 susceptible walkers remaining with everyone traced. When the alerted walkers' radius was reduced by half, in our simulation 32 susceptible walkers remained when no tracing was implemented as opposed to 276 susceptible walkers remaining when all walkers were traced. And when the walkers' radius remained unchanged when alerted (so the only

change was that they stopped moving), 28 walkers remained susceptible with no tracing as opposed to 179 walkers remaining susceptible with complete tracing. Figure 2 shows the difference in SIR curves between two scenarios in which the radius was reduced to 0.

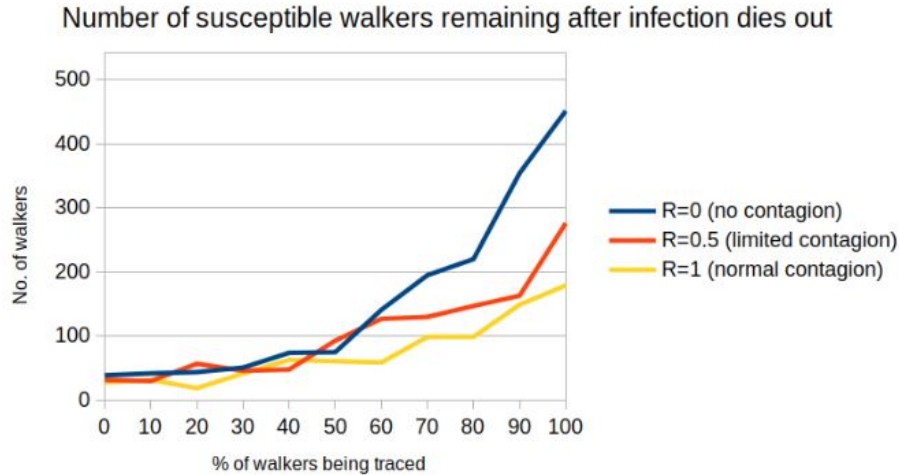


Figure 1: The above graph shows the number of walkers that remained susceptible until the end of a simulation on a square board of side 400 units depending on the percentage of walkers that were traced. R is the factor by which the walker's radius of infection is multiplied when they are alerted to isolate.

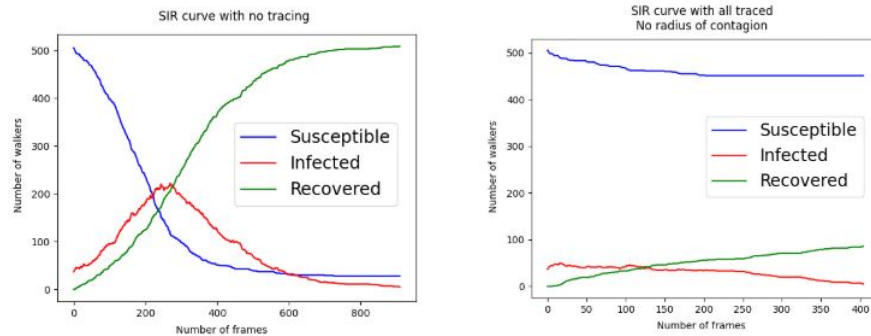


Figure 2: Here are two SIR curves obtained from running the simulation on a square board of size 400 units; the figure on the left corresponds to no traced walkers and the figure on the right corresponds to all walkers being traced and reducing their radius to 0 when alerted.

5 Conclusions

From our results above, a clear trend is visible: the greater the proportion of the population that is being traced, the greater the number of agents that don't experience the virus. We were not able to

run the simulation many times for each set of parameters we investigated, due to time constraints, so certain deviations in the numbers can be attributed to error. However, the number of walkers remaining susceptible increased by an entire order of magnitude when all walkers were traced as opposed to no tracing. Thus, it seems reasonable to conjecture that this is due to the effect of tracing rather than random chance.

It can be noted that the effects of tracing only become visible when at least 60 percent of the population is being traced, and in the case that the radius of alerted walkers is not reduced, visible changes only really occur past the 80 percent mark. In real life it could be difficult to get the majority of the population to comply with regulations, which severely limits the effectiveness of tracing.

As can be seen from the two SIR curves, not only do more walkers remain susceptible by the end of the model if tracing is implemented, but the peak of the number of infected walkers is also much lower. We were not able to investigate this result further, but suppose that this observation would be important in a real life situation as it strongly suggests that self isolating can help reduce the burden on health services by reducing overcrowding in hospitals. In our model there was no distinction between recovering from the virus and dying from it as both states lead to a walker not being susceptible to infection, but given greater time this would be an interesting topic to explore.

6 Philosophical reflections

As well as creating the model and drawing conclusions from it, we wanted to consider how we might justify the decisions we made when creating the model and when interpreting the results from a philosophical point of view. Specifically, what justification do we have in drawing conclusions about the world from this thing we have made? We explore this below.

Firstly, it is important to note why it is necessary to consider this question. A model, by its nature, is akin to the environment it is made to describe, but not identical to it. As such, there is always an inductive risk, an uncertainty concerning the applicability of information gained from the model to the real world, which arises from the fact that a model will never be truly symmetrical with the real world; it will only share certain features with reality. However, that element is more present in some models than others. Different models can be thought to rest at different points on the line between sharing a weak similarity and sharing an entire identity with their subject. Many philosophers of science hold that the closer it is to identity, the stronger the inferences that one can derive from it will be. That being said, no matter how similar a model is to its target system, one is always engaged in some kind of epistemic risk when reasoning in the realm of the model due to the nature of inductive reasoning.

This idea of an inductive gap is present in all areas of science, in different ways. However, values in science can be used to bridge this gap. Scientists must make complex decisions during their work which influences their findings. For example, some scientists would value a simple theory over a complex one (most likely, for example, in physics), whereas some find more value in the reverse (such as in ecology). A less controversial example could be internal consistency. By considering their adherence to these values, we can see how some theories are more likely to be epistemically useful than others.

In terms of making our model sufficiently representative, we chose to create an Agent Based Model because we wanted to represent how people behave. One feature of an ABM is that each individ-

ual object, or agent, bases its behaviour on the limited information available to it. Although we, as modellers, can gather statistical data about the model as a whole, this data is not available in real time to individual agents and cannot inform their behaviour. This is our main inductive justification. In other words, the conclusions that we draw from this model we justify in virtue of their similarity to something that's real, i.e., how humans behave. The symmetry between our model and the real world allows us to extrapolate information from the behaviour of the model to the probable behaviour of the real world.

Ultimately, the power of a model is tied to how it can be used as a space for surrogate reasoning about the real world. In other words, a model's epistemic utility is based on its functionality. We view our model as epistemically valuable because it is both representative and useful. We have made this model in order to create a space for 'surrogate reasoning'; in other words, we can derive information about the world through considering our model, in virtue of our model being similar to the world (representative), and producing results that we can observe (useful). The utility of the model is crucial; we can induce from our model valuable information pertaining to how societies can minimise infection risk by introducing track and trace.

Acknowledgements

We wish to thank our mentors for the Virtual ISSI programme: Charlotte, Pau and Joel for their dedicated support and amazing teaching throughout our project. We would also like to thank Adam and Adi for organising the programme, without whom none of this could have occurred.