Reinforcement Learning for Optimization of COVID-19 Mitigation Policies

Varun Kompella* 1, Roberto Capobianco* 1, 2, Stacy Jong 3, Jonathan Browne 3, Spencer Fox 3, Lauren Meyers 3, Peter Wurman 1, Peter Stone 1, 3

1 Sony AI
2 Sapienza University of Rome
3 The University of Texas at Austin

* Joint First Authors, varun.kompella@sony.com, roberto.capobianco@sony.com

Abstract

The year 2020 has seen the COVID-19 virus lead to one of the worst global pandemics in history. As a result, governments around the world are faced with the challenge of protecting public health, while keeping the economy running to the greatest extent possible. Epidemiological models provide insight into the spread of these types of diseases and predict the effects of possible intervention policies. However, to date, the even the most data-driven intervention policies rely on heuristics. In this paper, we study how reinforcement learning (RL) can be used to optimize mitigation policies that minimize the economic impact without overwhelming the hospital capacity. Our main contributions are (1) a novel agent-based pandemic simulator which, unlike traditional models, is able to model fine-grained interactions among people at specific locations in a community; and (2) an RL-based methodology for optimizing fine-grained mitigation policies within this simulator. Our results validate both the overall simulator behavior and the learned policies under realistic conditions.

1 Introduction

Motivated by the devastating COVID-19 pandemic, much of the scientific community, across numerous disciplines, is currently focused on developing safe, quick, and effective methods to prevent the spread of biological viruses, or to otherwise mitigate the harm they cause. These methods include vaccines, treatments, public policy measures, economic stimuli, and hygiene education campaigns. Governments around the world are now faced with high-stakes decisions regarding which measures to enact at which times, often involving trade-offs between public health and economic resiliency. When making these decisions, governments often rely on epidemiological models that predict and project the course of the pandemic.

The premise of this paper is that the challenge of mitigating the spread of a pandemic while maximizing personal freedom and economic activity is fundamentally a sequential decision-making problem: the measures enacted on one day affect the challenges to be addressed on future days. As such, modern reinforcement learning (RL) algorithms are well-suited to optimize government responses to pandemics. For such learned policies to be relevant, they must be trained within an epidemiological model that accurately simulates the spread of the pandemic, as well as the effects of government measures. To the best of our knowledge, none of the existing epidemiological simulations have the resolution to allow reinforcement learning to explore the regulations that governments are currently struggling with.

Motivated by this, our main contributions are:

1. The introduction of PANDEMICSIMULATOR, a novel open-source1 agent-based simulator that models the interactions between individuals at specific locations within a community. Developed in collaboration between AI researchers and epidemiologists (the co-authors of this paper), PANDEMICSIMULATOR models realistic effects such as testing with false positive/negative rates, imperfect public adherence to social distancing measures, contact tracing, and variable spread rates among infected individuals. Crucially, PANDEMICSIMULATOR models community interactions at a level of detail that allows the spread of the disease to be an emergent property of people’s behaviors and the government’s policies. An interface with OpenAI Gym (Brockman et al. 2016) is provided to enable support for standard RL libraries;

2. A demonstration that a reinforcement learning algorithm can indeed identify a policy that outperforms a range of reasonable baselines within this simulator;

3. An analysis of the resulting learned policy, which may provide insights regarding the relative efficacy of past and potential future COVID-19 mitigation policies.

While the resulting policies have not been implemented in any real-world communities, this paper establishes the potential power of RL in an agent-based simulator, and may serve as an important first step towards real-world adoption.

The remainder of the paper is organized as follows. We first discuss related work and then introduce our simulator in Section 3. Section 4 presents our reinforcement learning setup, while results are reported in Section 5. Finally, Section 6 reports some conclusions and directions for future work.

1https://github/sonyAI/PandemicSimulator
2 Related Work

Epidemiological models differ based on the level of granularity in which they track individuals and their disease states. “Compartmental” models group individuals of similar disease states together, assuming all individuals within a specific compartment to be homogeneous, and only track the flow of individuals between compartments (Tolles and Luong 2020). While relatively simplistic, these models have been used for decades and continue to be useful for both retrospective studies and forecasts as were seen during the emergence of recent diseases (Rivers and Scarpino 2018; Metcalf and Lessler 2017; Cobey 2020).

However, the commonly used macroscopic (or mass-action) compartmental models are not appropriate when outcomes depend on the characteristics of heterogeneous individuals. In such cases, network models (Bansal, Grenfell, and Meyers 2007; Liu et al. 2018; Khadilkar, Ganu, and Seetharam 2020) and agent-based models (Grefenstette et al. 2013; Del Valle, Mniszewski, and Hyman 2013; Aleta et al. 2020) may be more useful predictors. Network models encode the relationships between individuals as static connections in a contact graph along which the disease can propagate. Conversely, agent-based simulations, such as the one introduced in this paper, explicitly track individuals, their current disease states, and their interactions with other agents over time. Agent-based models allow one to model as much complexity as desired—even to the level of simulating individual people and locations as we do—and thus enable one to model people’s interactions at offices, stores, schools, etc. Because of their increased detail, they enable one to study the hyper-local interventions that governments consider when setting policy. For instance, Larremore et al. (2020) simulate the SARS-CoV-2 dynamics both through a fully-mixed mass-action model and an agent-based model representing the population and contact structure of New York City.

PANDEMICSIMULATOR has the level of details needed to allow us to apply RL to optimize dynamic government intervention policies (sometimes referred to as “trigger analysis” e.g. Duque et al. 2020). RL has been applied previously to several mass-action models (Libin et al. 2020; Song et al. 2020). These models, however, do not take into account individual behaviors or any complex interaction patterns. The work that is most closely related to our own includes both the SARS-CoV-2 epidemic simulators from Hoertel et al. (2020) and Aleta et al. (2020), which model individuals grouped into households who visit and interact in the community. While their approach builds accurate contact networks of real populations, it doesn’t allow us to model how the contact network would change as the government intervenes. Xiao et al. (2020) construct a detailed, pedestrian level simulation that simulates transmission indoors and studies three types of interventions. Liu (2020) presents a microscopic approach to model epidemics, which can explicitly consider the consequences of individuals’ decisions on the spread of the disease. Multi-agent RL is then used to let individual agents learn to avoid infections.

For any model to be accepted by real-world decision-makers, they must be provided with a reason to trust that it accurately models the population and spread dynamics in their own community. For both mass-action and agent-based models, this trust is typically best instilled via a model calibration process that ensures that the model accurately tracks past data. For example, Hoertel et al. (2020) perform a calibration using daily mortality data until 15 April. Similarly, Libin et al. (2020) calibrate their model based on the symptomatic cases reported by the British Health Protection Agency for the 2009 influenza pandemic. Aleta et al. (2020), instead, only calibrate the weights of intra-layer links by means of a rescaling factor, such that the mean number of daily effective contacts in that layer matches mean number of daily effective contacts in the corresponding social setting. While not a main focus of our research, we have taken initial steps to demonstrate that our model can be calibrated to track real-world data, as described in Section 3.

3 PandemicSimulator: A COVID-19 Simulator

The functional blocks of PANDEMICSIMULATOR, shown in Figure 1, are:

- **locations**, with properties that define how people interact within them;
- **people**, who travel from one location to another according to individual daily schedules;
- an **infection model** that updates the infection state of each person;
- an optional **testing strategy** that imperfectly exposes the infection state of the population;
- an optional **contact tracing** strategy that identifies an infected person’s recent contacts;
- a **government** that makes policy decisions.

The simulator models a day as 24 discrete hours, with each person potentially changing locations each hour. At the end of a day, each person’s infection state is updated. The government interacts with the environment by declaring **regulations**, which impose restrictions on the people and locations. If the government activates testing, the simulator identifies a set of people to be tested and (imperfectly) reports their infection state. If contact tracing is active, each person’s contacts from the previous days are updated. The updated perceived infection state and other state variables are returned as an observation to the government. The process iterates as long as the infection remains active. The fol-
lowing subsections describe the functional blocks of the simulator in greater detail.\footnote{We relegate some implementation details to an appendix at https://arxiv.org/pdf/2010.10560.pdf.}

**Locations**

Each location has a set of attributes that specify when the location is open, what roles people play there (e.g. worker or visitor), and the maximum number of people of each role. These attributes can be adjusted by regulations, such as when the government determines that businesses should operate at half capacity. Non-essential locations can be completely closed by the government. The location types used in our experiments are homes, hospitals, schools, grocery stores, retail stores, and hair salons. The simulator provides interfaces to make it easy to add new location types.

One of the advantages of an agent-based approach is that we can more accurately model variations in the way people interact in different types of locations based on their roles. The base location class supports workers and visitors, and defines a contact rate, $b_{loc}$, as a 3-tuple $(x, y, z) \in [0, 1]^3$, where $x$ is the worker-worker rate, $y$ is the worker-visitor rate, and $z$ is the visitor-visitor rate. These rates are used to sample interactions every hour in each location to compute disease transmissions. For example, consider a location that has a contact rate of $(0.5, 0.3, 0.4)$ and 10 workers and 20 visitors. In expectation, a worker would make contact with 5 co-workers and 6 visitors in the given hour. Similarly, a visitor would be expected to make contact with 3 workers and 8 other visitors. Refer to our supplementary material (Appendix A, Table 1) for a listing of the contact rates and other parameters for all location types used in our experiments.

The base location type can be extended for more complex situations. For example, a hospital adds an additional role (critically sick patients), a capacity representing ICU beds, and contact rates between workers and patients.

**Population**

A person in the simulator is an automaton that has a state and a person-specific behavior routine. These routines create person-to-person interactions throughout the simulated day and induce dynamic contact networks.

Individuals are assigned an age, drawn from the distribution of the US age demographics, and are randomly assigned to be either high risk or of normal health. Based on their age, each person is categorized as either a minor, a working adult or a retiree. Working adults are assigned to a work location, and minors to a school, which they attend 8 hours a day, five days a week. Adults and retirees are assigned favorite hair salons which they visit once a month, and grocery and retail stores which they visit once a week. Each person has a compliance parameter that determines the probability that the person flouts regulations each hour.

The simulator constructs households from this population such that 15\% house only retirees, and the rest have at least one working adult and are filled by randomly assigning the remaining children, adults, and retirees. To simulate informal social interactions, households may attend social events twice a month, subject to limits on gathering sizes.

At the end of each simulated day, the person’s infection state is updated through a stochastic model based on all of that individual’s interactions during the day (see next section). Unless otherwise prescribed by the government, when a person becomes ill they follow their routine. However, even the most basic government interventions require sick people to stay home, and at-risk individuals to avoid large gatherings. If a person becomes critically ill, they are admitted to the hospital, assuming it has not reached capacity.

**SEIR Infection Model**

**PandemicSimulator** implements a modified SEIR (susceptible, exposed, infected, recovered) infection model, as shown in Figure 2. See supplemental Appendix A, Table 2 for specific parameter values and the transition probabilities of the SEIR model. Once exposed to the virus, an individual’s path through the disease is governed by the transition probabilities. However, the transition from the susceptible state ($S$) to the exposed state ($E$) requires a more detailed explanation.

At the beginning of the simulation, a small, randomly selected set of individuals seeds the pandemic in the latent non-infectious, exposed state ($E$). The rest of the population starts in $S$. The exposed individuals soon transition to one of the infectious states and start interacting with susceptible people. For each susceptible person $i$, the probability they become infected on a given day, $P_{i}^{S \rightarrow E}(day)$, is calculated based on their contacts with infectious people that day.

$$
P_{i}^{S \rightarrow E}(day) = 1 - \prod_{t=0}^{23} P_{i}^{S \rightarrow E}(t) \tag{1}
$$

where $P_{i}^{S \rightarrow E}(t)$ is the probability that person $i$ is not infected at hour $t$. Whether a susceptible person becomes infected in a given hour depends on whom they come in contact with. Let $C_i^l(t) = \{p \sim N_j(t)| p \in N_{i}^{ml}(t)\}$ be the set of infected contacts of person $i$ in location $j$ at hour $t$ where $N_j(t)$ is the set of infected persons in location $j$ at time $t$, $N_j(t)$ is the set of all persons in $j$ at time $t$, and $b_j$ is a hand-set contact rate for $j$. To model the variations in how easily individuals spread the disease, each individual $k$ has an infection spread rate, $a_k \sim N^{\text{bounded}}(a, \sigma)$ sampled from a bounded Gaussian distribution. Accordingly,

$$
P_{i}^{S \rightarrow E}(t) = \prod_{k \in C_i^l(t)} (1 - a_k) \tag{2}
$$

**Testing and Contact Tracing**

**PandemicSimulator** features a testing procedure to identify positive cases of COVID-19. We do not model concomitant illnesses, so every critically sick or dead person is assumed to have tested positive. Non-symptomatic and symptomatic individuals—and individuals that previously tested positive—get tested all at different configurable rates. Additionally, we model false positive and false negative test
around the world. These parameters can also be used to customize the simulator to match a specific community. A discussion of our calibration process and the values we chose to model COVID-19 are discussed in Appendix A.

4 RL for Optimization of Regulations

An ideal solution to minimize the spread of a new disease like COVID-19 is to eliminate all non-essential interactions and quarantine infected people until the last infected person has recovered. However, the window to execute this policy with minimal economic impact is very small. Once the disease spreads widely this policy becomes impractical and the potential negative impact on the economy becomes enormous. In practice, around the world we have seen a strict lockdown followed by a gradual reopening that attempts to minimize the growth of the infection while allowing partial economic activity. Because COVID-19 is highly contagious, has a long incubation period, and large portions of the infected population are asymptomatic, managing the reopening without overwhelming healthcare resources is challenging. In this section, we tackle this sequential decision making problem using reinforcement learning (RL; Sutton and Barto 2018) to optimize the reopening policy.

To define an RL problem we need to specify the environment, observations, actions, and rewards.

Environment: The agent-based pandemic simulator PANDEMICSIMULATOR is the environment.\(^3\)

Actions: The government is the learning agent. Its goal is to maximize its reward over the horizon of the pandemic. Its action set is constrained to a pool of escalating stages, which it can either increase, decrease, or keep the same when it takes an action. Refer to Appendix A, Table 3 for detailed descriptions of the stages.

Observations: At the end of each simulated day, the government observes the environment. For the sake of realism, the infection status of the population is partially observable, accessible only via statistics reflecting aggregate (noisy) test results and number of hospitalizations.\(^4\)

Rewards: We designed our reward function to encourage the agent to keep the number of persons in critical condition \((n^c)\) below the hospital’s capacity \((C^\text{max})\), while keeping the economy as unrestricted as possible. To this end, we use a reward that is a weighted sum of two objectives:

\[
 r = a \max \left( \frac{n^c - C^\text{max}}{C^\text{max}}, 0 \right) + b \frac{\text{stage}^\text{p}}{\max_j \text{stage}^\text{p}_j} \quad (3)
\]

where \(\text{stage} \in [0, 4]\) denotes one of the 5 stages with stage\(_4\) being the most restrictive. \(a, b \) and \(p\) are set to \(-0.4, -0.1\) and \(1.5\), respectively, in our experiments. To discourage frequently changing restrictions, we also use a small shaping

\(^3\)For the purpose of our experiments, we assume no vaccine is on the horizon and that survival rates remain constant. In practice, one may want to model the effect of improving survival rates as the medical community gains experience treating the virus.

\(^4\)The simulator tracks ground truth data, like the number of people in each infection state, for evaluation and reporting.
Figure 3: A single run of the simulator with no government restrictions, showing (a) the true global infection summary, (b) the perceived infection state, and (c) the number of people in critical condition over time.

shows the metrics observed by the government through the lens of testing and hospitalizations. This plot illustrates how the government sees information that is both an underestimate of the penetration and delayed in time from the true state. Finally, Figure 3(c) shows that the number of people in critical condition goes well above the maximum hospital capacity (denoted with a yellow line) resulting in many people being more likely to die. The goal of a good reopening policy is to keep the red curve below the yellow line, while keeping as many businesses open as possible.

Figure 4 shows plots of our infection metrics averaged over 30 randomly seeded runs. Each row in Figures 4(a-o) shows the results of executing a different (constant) regulation stage (after a short initial S0 phase), where S4 is the most restrictive and S0 is no restrictions. As expected, Figures 4(p-r) show that the infection peaks, critical cases and number of deaths are all lower for more restrictive stages. One way of explaining the effects of these regulations is that the government restrictions alter the connectivity of the contact graph. For example, in the experiments above, under stage 4 restrictions there are many more connected components in the resulting contact graph than in any of the other 4 cases. See Appendix A for details of this analysis.

Higher stage restrictions, however, have increased socioeconomic costs (Figure 4(s); computed using the second objective in Eq. 3). Our RL experiments illustrate how these competing objectives can be balanced.

A key benefit of PANDEMICSIMULATOR’s agent-based approach is that it enables us to evaluate more dynamic policies than those described above. In the remainder of this section we compare a set of hand-constructed policies, examine (approximations) of two real country’s policies, and study the impact of contact tracing. In Appendix A we also provide an analysis of the model’s sensitivity to its parameters. Finally, we demonstrate the application of RL to construct dynamic policies that achieve the goal of avoiding exceeding hospital capacity while minimizing economic costs. As in Figure 4, throughout this section we report our results using plots that are generated by executing 30 simulator runs with fixed seeds. All our experiments were run on a single core, using an Intel i7-7700K CPU @ 4.2GHz with 32GB of RAM.

**Benchmark Policies**

To serve as benchmarks, we defined three heuristic and two policies inspired by real governments’ approaches to managing the pandemic.

- **S0-4-0**: Using this policy, the government switches from stage 0 to 4 after reaching a threshold of 10 infected persons. After 30 days, it switches directly back to stage 0;
- **S0-4-0-FI**: The government starts like S0-4-0, but after 30 days it executes a fast, incremental (FI) return to stage 0, with intermediate stages lasting 5 days;

---

5 PANDEMICSIMULATOR can easily handle larger experiments at the cost of greater time and computation. Informal experiments showed that results from a population of 1k are generally consistent with results from a larger population when all other settings are the same (or proportional). Refer to Table 7 in the appendix for simulation times for 1k and 10k population environments.

6 Such as at https://tinyurl.com/y3pjthyz

---

7 In this paper, we use the word “policy” to mean a function from state of the world to the regulatory action taken. It represents both the government’s policy for combating the pandemic (even if heuristic) and the output of an RL optimization.
To validate PANDEMICSIMULATOR’s ability to model testing and contact tracing we compare several strategies with different testing rates and contact horizons. We consider daily testing rates of 0.02, 0.3, and 1.0 (where 1.0 represents the extreme case of everyone being tested every day) and contact tracing histories of 0, 2, 5, or 10 days. For each condition, we ran the experiments with the same 30 random seeds. The full results appear in Appendix A.

Not surprisingly, contact tracing is most beneficial with higher testing rates and longer contact histories because more testing finds more infected people and the contact tracing is able to encourage more of that person’s contacts to stay home. Of course, the best strategy is to test every person every day and quarantine anyone who tests positive. Unfortunately, this strategy is impractical except in the most isolated communities. Although this aggressive strategy often stamps out the disease, the false-negative test results sometimes allow the infection to simmer below the surface and spread very slowly through the population.

**Optimizing Reopening using RL**

A major design goal of PANDEMICSIMULATOR is to support optimization of re-opening policies using RL. In this section, we test our hypothesis that a learned policy can outperform the benchmark policies. Specifically, RL optimizes a policy that (a) is adaptive to the changing infection state, (b) keeps the number of critical patients below the hospital threshold, and (c) minimizes the economic cost.

We ran experiments using the 5-stage regulations defined in Table 3 (Appendix A); trained the policy by running RL optimization for roughly 1 million training steps; and evaluated the learned policies across 30 randomly seeded initial conditions. Figures 6(a-f) show results comparing our best heuristic policy (S0-4-0-GI) to the learned policy. The learned policy is better across all metrics as shown in Fig-

---

Footnotes:

8https://tinyurl.com/y57yq2x7; https://tinyurl.com/y34egdeg
9https://tinyurl.com/y3cepy3m
Further, we can see how the learned policy reacts to the state of the pandemic; Figure 6(f) shows different traces through the regulation space for 3 of the trials. The learned policy briefly oscillates between Stages 3 and 4 around day 40. To minimize such oscillations, we evaluated the policy at an action frequency of one action every 3 days (bi-weekly; labeled as Eval_w3) and every 7 days (weekly; labeled as Eval_w7). Figure 6(p) shows that the bi-weekly variant performs well, while making changes only once a week slightly reduces the reward. To test robustness to scaling, we also evaluated the learned policy (with daily actions) in a town with a population of 10,000 (Eval_10k) and found that the results transfer well. This success hints at the possibility of learning policies quickly even when intending to transfer them to large cities.

This section presented results on applying RL to optimize reopening policies. An interesting next step would be to study and explain the learned policies as simpler rule-based strategies to make it easier for policy makers to implement. For example, in Figure 6(l), we see that the RL policy waits at stage 2 before reopening schools to keep the second wave of infections under control. Whether this behavior is specific to school reopening is one of many interesting questions that this type of simulator allows us to investigate.

6 Conclusion

Epidemiological models aim at providing predictions regarding the effects of various possible intervention policies that are typically manually selected. In this paper, instead, we introduce a reinforcement learning methodology for optimizing adaptive mitigation policies aimed at maximizing the degree to which the economy can remain open without overwhelming the local hospital capacity. To this end, we implement an open-source agent-based simulator, where pandemics can be generated as the result of the contacts and interactions between individual agents in a community. We analyze the sensitivity of the simulator to some of its main parameters and illustrate its main features, while also showing that adaptive policies optimized via RL achieve better performance when compared to heuristic policies and policies representative of those used in the real world.

While our work opens up the possibility to use machine
learning to explore fine-grained policies in this context, \textsc{pandemicsimulator} could be expanded and improved in several directions. One important direction for future work is to perform a more complete and detailed calibration of its parameters against real-world data. It would also be useful to implement and analyze additional testing and contact tracing strategies to contain the spread of pandemics.

References


