Reinforcement Learning Enhances the Experts: Large-scale COVID-19 Vaccine Allocation with Multi-factor Contact Network

Qianyue Hao
Department of Electronic Engineering, Tsinghua University
Beijing, China

Wenzhen Huang
Department of Electronic Engineering, Tsinghua University
Beijing, China

Fengli Xu
Department of Sociology, University of Chicago
Chicago, The USA

Kun Tang
Vanke School of Public Health, Tsinghua University
Beijing, China

Yong Li*
Department of Electronic Engineering, Tsinghua University
Beijing, China

ABSTRACT

In the fight against the COVID-19 pandemic, vaccines are the most critical resource but are still in short supply around the world. Therefore, efficient vaccine allocation strategies are urgently called for, especially in large-scale metropolis where uneven health risk is manifested in nearby neighborhoods. However, there exist several key challenges in solving this problem: (1) great complexity in the large scale scenario adds to the difficulty in experts' vaccine allocation decision making; (2) heterogeneous information from all aspects in the metropolis' contact network makes information utilization difficult in decision making; (3) when utilizing the strong decision-making ability of reinforcement learning (RL) to solve the problem, poor explainability limits the credibility of the RL strategies. In this paper, we propose a reinforcement learning enhanced experts method. We deal with the great complexity via a specially designed algorithm aggregating blocks in the metropolis into communities and we hierarchically integrate RL among the communities and experts solution within each community. We design a self-supervised contact network representation algorithm to fuse the heterogeneous information for efficient vaccine allocation decision making. We conduct extensive experiments in three metropolis with real-world data and prove that our method outperforms the best baseline, reducing 9.01% infections and 12.27% deaths. We further demonstrate the explainability of the RL model, adding to its credibility and also enlightening the experts in turn.

CCS CONCEPTS

• **Computing methodologies** → **Reinforcement learning**; • **Applied computing** → **Life and medical sciences**.

KEYWORDS

Reinforcement learning; self-supervised representation learning; model explainability; COVID-19 pandemic; vaccine allocation.

Corresponding author. Email: liyong07@tsinghua.edu.cn.

KDD '22, August 14–18, 2022, Washington, DC, USA

ACM ISBN 978-1-4503-9385-0/22/08.
https://doi.org/10.1145/3534678.3542679

ACM Reference Format:
Qianyue Hao, Wenzhen Huang, Fengli Xu, Kun Tang, and Yong Li*. 2022. Reinforcement Learning Enhances the Experts: Large-scale COVID-19 Vaccine Allocation with Multi-factor Contact Network. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '22), August 14–18, 2022, Washington, DC, USA.* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3534678.3542679

1 INTRODUCTION

In the long-lasting fight against the COVID-19 pandemic, the vaccines now are and remain to be the most important tools in our human hands [8]. However, due to the limited supply [3] and inequitable distribution around the world [19], COVID-19 vaccines remain far less than enough to cover most of the people. Therefore, efficient strategies for allocating the currently available vaccines and maximizing the benefit are urgently called for. Besides, the virus tends to spread faster in the crowded metropolis [31] like metropolitan statistical areas (MSA[1]s), where there are thousands of census block groups (CBG[2]s) and millions of people. This indicates that efficient large-scale vaccine allocation strategies in the metropolis are especially important and necessary.

The problem of COVID-19 vaccine allocation has long been studied by public health experts. Strategies based on individual characteristics such as age [18] or health state [26] are widely proposed. While other guidelines focus on macro goals such as saving the most lives [23] or guaranteeing equality [24]. But it is difficult to turn these abstractive high-level guidances into practical strategies to implement in the real world metropolis. On the other hand, reinforcement learning (RL) for pandemic intervention has long been studied due to its strong ability in decision making, including efficient lockdown strategies [17, 20] and targeted border testing [1], etc. However, methods on the problem of vaccine allocation, especially in large-scale metropolis, are almost unexplored.

Despite the importance and necessity of efficient strategies for large-scale COVID-19 vaccine allocation in the metropolis, there exist several key challenges. (1) Though expert solutions considering transmission risk work well in many scenarios [10], the allocation complexity increases exponentially as the number of CBGs goes up in large-scale MSAs, making the traditional expert solutions less and less powerful. (2) Multi-factor contact network, the dynamic

[1]Geographical regions with a relatively high population density and close economic ties throughout the area.
[2]The smallest geographical unit for which the bureau publishes sample data.

bipartite network combined with census block groups (CBGs) and points of interest (POI[3]s), contains abundant information on demographic features and population mobility patterns about the MSAs, providing a strong basis for vaccine allocation decision making. However, the large amount of heterogeneous information also adds difficulties to efficient information utilization. Therefore, either traditional expert solutions or AI methods are only able to utilize a very limited part of the information without an efficient information representation method and thus cannot reach good enough performance. (3) When trying to exert the strong decision-making ability of RL for the problem, though well-trained RL agent has the capability to make efficient strategies, the decision-making process tends to be like a black box and lacks of explainability. These black-box decisions are not convincing to the experts as well as the public, especially on serious matters related to life and health, which limits their credibility and feasibility in the real world situation. Meanwhile, the RL methods will not improve the experts' knowledge at all due to the lack of explainability.

In this paper, we propose a reinforcement learning enhanced experts method to solve this problem in view of these challenges. To deal with the complexity in large-scale MSAs, we design a community division algorithm, aggregating the CBGs with similar mobility patterns and spatial adjacency into communities. Then we hierarchically allocate the vaccines among the communities via RL method and within each community via an expert solution, improving the efficiency of allocation strategies. In order to efficiently utilize the hidden information in the multi-factor contact network, we designed a self-supervised network representation method, which learns the required representations automatically, providing a strong basis for RL decision making. We conduct extensive experiments and targeted ablation studies in three MSAs with real-world data and the results clearly exhibit the good performance of our method. We further deeply analyze the trained RL model to uncover the key factors considered in the RL decision-making process and thus we add to the explainability of our method, making it more credible and feasible in the real world. Furthermore, the interpretable RL strategies not only enhance but also enlighten the experts, providing new knowledge on similar problems.

The main contributions of our work include:

- We propose a reinforcement learning enhanced experts method for solving the important yet difficult problem of large-scale COVID-19 vaccine allocation. Due to the optimized design, we propose more efficient strategies comparing pure expert solution, which enable allocating vaccines efficiently in large-scale MSAs with thousands of CBGs and millions of people.
- We design a self-supervised representation method for efficiently utilizing the information in the multi-factor contact network of MSAs, enhancing the performance. We conduct extensive experiments in various MSAs with real-world data and demonstrate that our method outperforms the best baseline, reducing 9.01% infections and 12.27% deaths.
- We analyze the explainability of the RL. Therefore, we uncover how RL makes efficient strategies, which adds to the credibility and real-world feasibility of RL strategies and also enlightens the future experts' decisions on vaccine allocation.

[3]Specific locations that someone may find useful or interesting.

2 PRELIMINARIES

2.1 Multi-factor Contact Network Data

In this paper, we mainly study the metropolitan statistical areas (MSAs), taking census block groups (CBGs) and points of interest (POIs) as the minimum studying units. First, we give a formal definition of the multi-factor contact network as follows:

Definition 1 (Multi-factor Contact Network). *In an MSA with M CBGs and N POIs. The multi-factor contact network is a dynamic bipartite network with M CBG nodes and N POI nodes over T time steps, denoted as $G[t], t = 1, 2, ...T$. Each CBG and POI node has time-invariant static features such as population age structure of each CBG, the floor space and average dwelling time of each POI, and time-changing dynamic features such as visiting fluid of each POI at each time step. The weight of edge between CBG m and POI n at time t is denoted as $e_{mn}[t] \in \mathbb{R}, m = 1, 2, ...M, n = 1, 2, ...N$, indicating the number of people in CBG m who visit POI n at time t.*

We conduct the following studies based on the real-world multi-factor contact networks data in three MSAs with the temporal resolution of one hour, which are captured by previous researchers from the SafeGraph open data and are available online [5]. More details of the multi-factor contact networks are shown in Section 3.3.

2.2 COVID-19 Pandemic Spreading Modeling

There exists plentiful works on modeling and simulating the COVID-19 pandemic spreading, providing convincing foundations for the downstream tasks [5, 6, 13]. In this paper, we mainly adopt the Behavior and Demography informed epidemic model (BD model) [7], which is an improvement of the well-known meta-population model [5]. We show the overview of this model in Figure 1.

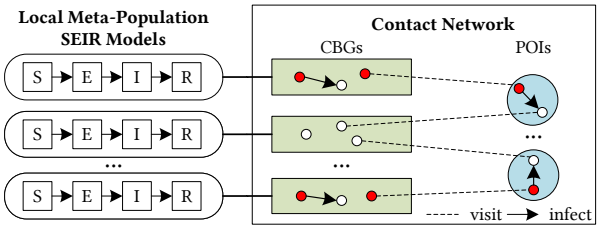


Figure 1: Overview of the Behavior and Demography informed epidemic model (BD model).

Briefly, the BD model characterizes Intra-CBG transmissions by maintaining local meta-population SEIR models with 4 states in each CBG, i.e., S for susceptible, E for exposed, I for infectious, and R for removed, where a certain proportion of I turn to be reported cases and are observable according to the testing capability. Besides, a proportion of R fall into deaths according to CBG specified infection-fatality rate (IFR) estimated by the population age structure of each CBG and age specified death risks, while others in R turn to recover. This model characterizes Inter-CBG transmissions based on the contact network among CBGs and POIs, which happen when S individuals visit POIs and encounter I individuals from other CBGs. The transmission probability in each POI is positively related to its average dwelling time and is inversely proportional to its floor space. The effect of COVID-19 vaccines in each CBG in the BD model is characterized as an equivalent reduction on the

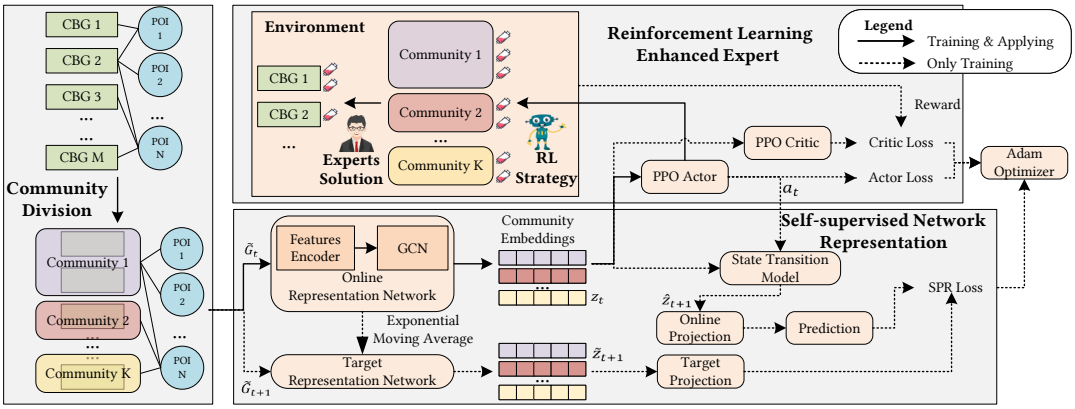


Figure 2: Overview of the whole system with community division algorithm, reinforcement learning enhanced experts method and self-supervised network representation.

corresponding infect rate, i.e., the probability for S individuals to be infected and turn into E in contact with I individuals. The reduction proportion equals to the percentage of vaccinated individuals in such CBG. Mentioning that we simplify the settings into obtaining 100% immunity after one dose of vaccine injection in this paper while real-world situations of two-injection vaccines or $< 100\%$ immunity can be considered by simply changing the parameters.

The accuracy of the BD model has been verified by previous researchers [7] by comparing the simulation results with the real-world situations in nine MSAs and resulting in a normalized root mean square error <0.5 on daily deaths number. Therefore, it can serve as our experimental platform and support the following study.

2.3 Problem Overview

Now we give a formal definition of our research problem:

Problem 1 (COVID-19 Vaccine Allocation). *In an MSA with M CBGs, given the corresponding multi-factor contact network over T time steps and the available number of COVID-19 vaccines $V[t], t = 1, 2, ...T$, at each time step, find out the number of vaccines allocated to each CBG at each time step $V_m[t], m = 1, 2, ...M, t = 1, 2, ...T$, minimizing the pandemic damage and ensuring $\sum_m V_m[t] \leq V[t]$.*

We focus on large-scale vaccine allocation in this paper, where large-scale means MSAs with large-scale population and CBGs, typically up to millions of people and thousands of CBGs. The quantitative metrics for the pandemic caused damage include:

- **Total Cases**: Total number of people who have ever been infected (recovered, dead, or still under infection) during the period. In the real world, only tested cases are counted.
- **Total Deaths**: Total number of deaths during the period.

It is obvious that to minimize the pandemic damage is to minimize the above metrics. When quantitatively evaluating the efficiency of various COVID-19 vaccine allocation strategies, we fix the vaccine number $V[t]$ and compare the decrease on the above two metrics contrasting the no vaccine scenario with different strategies. In this way, we can measure the efficiency of different vaccine allocation strategies in utilizing the same amount of vaccines to reduce the damage caused by the pandemic.

3 METHODS

3.1 System Overview

To solve the problem of large-scale COVID-19 vaccine allocation, we propose a reinforcement learning enhanced experts method with a self-supervised network representation mechanism. The overview of the system structure is shown in Figure 2.

First, we design a community division algorithm on the multi-factor contact network and aggregate the CBGs into communities (Section 3.2). Second, we utilize a self-supervised representation learning method, which is synergistically trained with the RL agent to obtain the embedding vectors of the multi-factor contact network (Section 3.3). Third, we design a two-layers hierarchical structure where the RL agent takes in the embedding vectors and allocates the vaccines among the communities, and then an expert solution is applied to allocate vaccines inside each community (Section 3.4). By considering CBGs in the same community as a whole in the first layer, we are able to greatly reduce the computation complexity of the RL agent. Mentioning that the pipeline from the multi-factor contact network data to the final vaccine allocation strategies in our method consists of the above three sections by order. Finally, we design a quantitative method for explainability analyses on the trained RL model and thus uncover the key factors in RL decision-making (Section 3.5). Our method is implemented with PyTorch and the source codes are available at https://github.com/KYHKL-H/RL-enhance-expert. All the hyper-parameters used in our implementation are summarized in Appendix A for reproducibility.

3.2 Community Division on Contact Network

To aggregate CBGs with similar POI visiting patterns into communities and reduce the computation complexity of the RL agent, we design a community division algorithm working on the multi-factor contact network. We first eliminate the POI nodes in the multi-factor contact network over all the time steps and get a new static network with only CBG nodes. The edge between two CBG nodes in the new network reflects how close the relation between these two CBGs is and the major principle is that the more people from these two CBGs encounter in the raw CBG-POI contact network, the closer the relation is. Then we perform the Fluid Communities algorithm [22] on the obtained new static network to

obtain the final community division result. We select this algorithm considering its performance and low computational cost. We show the detailed steps in Algorithm 1.

Algorithm 1 Community Division

Input: Multi factor CBG-POI contact network $G[t]$ with M CBGs, N POIs and edge weights $e_{mn}[t]$, encounter threshold ξ, average community size S
Output: Community division result
1: Initialize new static network G' with M nodes and no edge
2: **for** $t = 1, 2, ...T; m_1, m_2 = 1, 2, ...M, m_1 \neq m_2; n = 1, 2, ...N$ **do**
3: **if** $e_{m_1 n}[t] \times e_{m_2 n}[t] > \xi$ **then**
4: Add unweighted edge between nodes m_1 and m_2 into G' if the edge does not exists before
5: **end if**
6: **end for**
7: Apply Fluid Communities algorithm on G' with community number $K = \lceil M/S \rceil$ and obtain the community division result
8: **return** Community division result

The community division results are evaluated in Section 4.2. We verify that via our community division algorithm, CBGs in the same community not only have similar POIs visiting patterns but are also spatially adjacent, which ensures the feasibility of regarding them as a whole and applying union vaccine allocation strategies on the community level. In other words, spatial adjacency makes the real-world vaccine allocation, transportation, and distribution to CBGs within the same community greatly convenient.

3.3 Self-supervised Representation for Multi-factor Contact Network

3.3.1 Details of the Multi-factor Contact Network. The CBG and POI nodes in the multi-factor contact network have various static and dynamic features, aggregating information from multiple aspects. Also, the time-varying weighted edges describe the frequent encounters of people from different CBGs in the POIs. Therefore, the multi-factor contact network contains detailed descriptions of the hidden patterns of population dynamics and the COVID-19 pandemic spreading situation in the MSAs, providing a strong basis for the vaccine allocation decision making.

We adopt the following CBG and POI node features in the raw data into our multi-factor contact network:

- **CBG Age**: Static feature describing the population age structure of each CBG, which is represented as a vector consists of percentages of 23 age groups, denoted as $C_m^A \in [0, 1]^{23}$ for CBG m and $\sum_{dim}^{23} C_m^A = 1, \forall m$.
- **POI Visit**: Dynamic feature for the total number of people visiting each POI at each time step, where that of POI n at time step t is $P_n^V[t] = \sum_m e_{mn}[t]$.
- **POI Area**: Static feature describing the floor space of each POI, denoted as P_n^A for POI n. The smaller it is, the more crowded the POI tends to be and therefore the higher the COVID-19 transmission probability in such POI is.
- **POI Time**: Static feature describing the average visitors' dwelling time in each POI, denoted as P_n^T for POI n. The longer it is, the higher the COVID-19 transmission probability in such POI is.

Using the community division algorithm in Section 3.2, we obtain the community division result and here we aggregate the CBG nodes into community nodes and obtain the community-POI multi-factor contact network $\hat{G}[t], t = 1, 2, ...T$ from the original CBG-POI one $G[t]$. We denote the population size of each CBG as C_m^P and denote the index of the community each CBG belonging to as C_m^I. The community nodes feature **Community Age** for community $k, k = 1, 2, ...K$ is calculated as weighted average of C_m^A:

$$\hat{C}_k^A = \frac{1}{\sum_m \mathbb{1}[C_m^I = k] C_m^P} \sum_m \mathbb{1}[C_m^I = k] C_m^P C_m^A, \tag{1}$$

where $\mathbb{1}[x = y]$ is assigned to 1 if x equals to y, otherwise it is assigned to 0. The edge weight between community k and POI n is:

$$\hat{e}_{kn}[t] = \sum_m \mathbb{1}[C_m^I = k] e_{mn}[t], \tag{2}$$

and thus the POI nodes feature **POI Visit** is:

$$\hat{P}_n^V[t] = \sum_m e_{mn}[t] = \sum_k \hat{e}_{kn}[t]. \tag{3}$$

We set the changing frequency of our vaccine allocation strategies to one day. Therefore we temporally aggregate every 24 steps of the hourly community-POI multi-factor contact network $\hat{G}$ into one step in the daily network $\tilde{G}$, where the **Edge Weights** are:

$$\tilde{e}_{kn}[\tau] = \sum_{t=24\tau-23}^{24\tau} \hat{e}_{kn}[t], \tag{4}$$

and the POI node feature **POI Visit** is:

$$\tilde{P}_n^V[\tau] = \sum_{t=24\tau-23}^{24\tau} \sum_m e_{mn}[t] = \sum_{t=24\tau-23}^{24\tau} \sum_k \hat{e}_{kn}[t]. \tag{5}$$

We denote the cumulative number of reported cases, the cumulative number of deaths and remaining number of susceptible people in each community at time step t as $I_k[t]$, $D_k[t]$ and $S_k[t]$ and newly add the following two CBG dynamic features related to the COVID-19 pandemic spreading situation into $\tilde{G}$:

- **Community States**: Pandemic spreading situations in each community over the last 24 time steps, which includes three sub-features, i.e., S-State $\{S_k[t-23], ...S_k[t-1], S_k[t]\}$, I-State $\{I_k[t-23], ...I_k[t-1], I_k[t]\}$ and D-State $\{D_k[t-23], ...D_k[t-1], D_k[t]\}$ for community k at time $\tau = t/24$.
- **Community Diffs**: The difference between the current pandemic spreading situation and the situation one day before, which also includes three sub-features, i.e., S-Diff $S_k[t] - S_k[t-24]$, I-Diff $I_k[t] - I_k[t-24]$ and D-Diff $D_k[t] - D_k[t-24]$ for community k at time $\tau = t/24$.

These two dynamic features contain information about the pandemic spreading situation and thus are critical in our specific problem of vaccine allocation. The obtained multi-factor contact network $\tilde{G}$ contains abundant information about the corresponding MSA and serves as the environment state in the following steps.

3.3.2 Neural Network Structure. We show the detailed neural network structure for multi-factor contact network representation in Figure 3, which is corresponding to the Online and Target Representation Network in Figure 2. We set all non-linear activation functions between layers to be Leaky-ReLU function [33], i.e., $F(x) = max(x, \alpha x)$. The parameters in the fully connected layers are shared

among the communities and POIs respectively and we apply the normalization over each community and POI in each mini-batch, respectively. Also, we use the normalized edge weights to weigh the passing messages in the graph convolutional network (GCN) because of the intuition that a larger edge weight reflects a closer relation and thus requires a larger weight on the corresponding passing messages. We design three different vector normalization methods in the neural network for different types of data, i.e., dynamic norm, static norm, and edge weights norm, and the detailed mathematical formulations are shown in Appendix B.

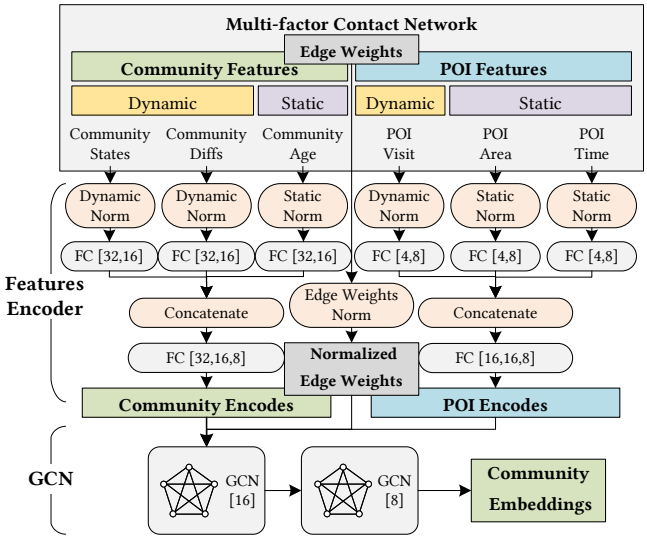


Figure 3: Neural network structure for multi-factor contact network representation. For instance, FC [4, 8] means two fully connected layers with 4 and 8 output units respectively.

3.3.3 Self-supervised Training Algorithm. To improve information utilization efficiency in decision making, we apply Self-Predictive Representations (SPR) [29] for self-supervised multi-factor contact network representation, which is designed for the fact that state representations to be predictive of future states given future actions are good representations. The detailed structure of the representation network is shown in Figure 3 and those of the auxiliary neural networks, i.e., state transition model, projection, and prediction are shown in Appendix C. During the training process, the parameters of the target neural networks θ_m are obtained from corresponding online neural networks θ_o via delayed synchronization method exponential moving average (EMA) as follows:

$$\theta_m \leftarrow \kappa\theta_m + (1-\kappa)\theta_o. \tag{6}$$

We implement one-step SPR and the outline of our self-supervised representation learning training algorithm is shown in Algorithm 2. The tuple (s, a, r, s') refers to the elements of state, action, reward, and next state in the Markov Decision Process (MDP) framework.

As we show in the algorithm, the contact network representation is synergistically trained with the RL agent, managing to learn good representations, especially of the space explored by the RL agent. Besides, the representations also contain the environment state transition patterns and thus can enhance the performance of RL.

Algorithm 2 Self-supervised Representation Learning Training

Input: Bath size N_0, training epoch E, EMA ratio κ and replay buffer size N_B
Output: Trained representation and RL neural network

Initialize online representation network f_o and projection g_o with θ_o, initialize target representation network f_m and projection g_m with $\theta_m \leftarrow \theta_o$
Initialize transition model h, predictor q and RL network ϕ
Initialize replay buffer B with size N_B
while Model Training **do**
  Collect (s, a, r, s') with θ_o and ϕ to fill B, $s = \tilde{G}_t, s' = \tilde{G}_{t+1}$
  for *batch count* $= 1, 2, ...\lfloor E \times N_B/N_0 \rfloor$ **do**
    Sample $\{(s_i, a_i, r_i, s_i') \mid i = 1, 2, ...N_0\} \sim B$
    $z_t \leftarrow f_o(\{s_i\})$, $\tilde{z}_{t+1} \leftarrow f_m(\{s_i'\})$
    $\hat{z}_{t+1} \leftarrow h(z_t, \{a_i\})$
    $\hat{y} \leftarrow q(g_o(\hat{z}_{t+1}))$, $\tilde{y} \leftarrow g_m(\tilde{z}_{t+1})$
    Loss $l \leftarrow -\lambda \frac{1}{N_0}(\frac{\hat{y}}{\|\hat{y}\|_2})^T(\frac{\tilde{y}}{\|\tilde{y}\|_2})$+RL Loss$(\{(s_i, a_i, r_i, s_i')\})$
    Update θ_o and ϕ minimizing l
    $\theta_m \leftarrow \kappa\theta_m + (1-\kappa)\theta_o$
  end for
  Empty B
end while
return θ_o and ϕ

3.4 Reinforcement Learning Enhanced Experts

To solve the difficult problem of COVID-19 vaccine allocation in large-scale scenario, where the great complexity limits the performance of pure expert solutions, we design a two-layer hierarchical structure, namely the RL enhanced experts method. We utilize the strong ability of RL in serial decision-making to allocate the limited number of vaccines among the divided communities and then set an expert solution among the CBGs within each community, where the scale and complexity are already largely reduced.

For the RL layer, we implement the widely used proximal policy optimization (PPO) algorithm [28], which consists of an actor-critic structure. The actor takes in the obtained community embeddings and outputs two K-dimensional vectors denoted as $\mu = [\mu_1, \mu_2, ...\mu_K]^T$ and $\sigma = [\sigma_1, \sigma_2, ...\sigma_K]^T$, corresponding to the K communities. Then we build a K-dimensional multivariate normal distribution based on μ, σ and then sample a K-dimensional action vector x from the distribution $x = [x_1, x_2, ...x_K]^T \sim N(\mu, \Sigma)$, where the covariance matrix $\Sigma = diag(\sigma_1^2, \sigma_2^2, ...\sigma_K^2)$ is non-negative definite. Then we calculate the proportion of vaccines allocated to community k, denoted as p_k via the softmax function $p_k = \frac{e^{x_k}}{\sum_i e^{x_i}}$.

During the training process, we set the reward function to be the opposite number of the increasing ratio of cases and deaths in the last 24 hours:

$$r_t = -\frac{\Sigma_k(C_k[t] - C_k[t-24])}{\Sigma_k C_k[t-24] + \epsilon_r} - \frac{\Sigma_k(D_k[t] - D_k[t-24])}{\Sigma_k D_k[t-24] + \epsilon_r}, \tag{7}$$

i.e., the more new cases and deaths, the lower reward. The critic network, denoted as V, serves as the state value function, and we calculate the loss of it using **SmoothL1** loss function as follows:

$$l_c(\theta_c) = \mathbf{E}_t[\mathbf{SmoothL1}(\gamma V_{\theta_c}(s_{t+1}) + r_t, V_{\theta_c}(s_t))], \tag{8}$$

where γ is the decay rate of long term reward. Mentioning that we minimize $l_c(\theta_c)$ only with gradient from $V_{\theta_c}(s_t)$. And we calculate the loss of actor network π through the typical PPO loss:

$$l_a(\theta_a) = -\mathbf{E}_t[Min[R_t(\theta_a)\hat{A}_t, Clip(r_t(\theta_a), 1-\epsilon, 1+\epsilon)\hat{A}_t]], \quad (9)$$

where $Clip(x,a,b) = Min[Max[x,a],b]$, $R_t(\theta_a) = \frac{\pi_{\theta_a}(a_t|s_t)}{\pi_{\theta_{a,old}}(a_t|s_t)}$, and the advantage function $\hat{A}_t$ is estimated via generalized advantage estimation (GAE) [27] as follows:

$$\hat{A}_t = \gamma V_{\theta_c}(s_{t+1}) + r_t - V_{\theta_c}(s_t). \quad (10)$$

Therefore, the RL Loss item in Algorithm 2 equals to $l_c(\theta_c) + l_a(\theta_a)$. More details including neural network structures of the actor and critic are shown in Appendix D.

For the expert solution layer, as suggested by WHO [21], populations with a higher risk of being infected are supposed to have a higher priority to the vaccines. Therefore, we further allocate the vaccines in a certain community to each CBG in proportion to the number of newly reported cases in the last 24 hours, because people in CBGs with more new cases are more likely to be infected.

3.5 Explainability Analyses on the RL Model

Inspired by previous work [12], we design a method to analyze which factors in the multi-factor contact network play the most roles in RL decision making. Thus we can shed light on the explainability of RL strategies and enlighten the experts in turn. With the trained model $\mathcal{M}$ and the input factors $\{f_1, f_2, ...\}$, which are tensors with different dimensions, we mask one of the factors by replacing all its elements with its mean value and get $\hat{f_i}$. Then we analyse the importance of f_i, denote as ψ_i by calculating the absolute change on the outputs before and after masking it as follows:

$$\psi_i = | \mathcal{M}(\{f_j\}) - \mathcal{M}(\{f_j \mid j \neq i\} \cup \{\hat{f_i}\}) |, \quad (11)$$

where larger ψ_i means that the outputs of $\mathcal{M}$ are affected more by f_i, i.e., f_i plays a more important role.

4 EXPERIMENTS

4.1 Experimental Settings

To evaluate the performance of our method in solving real-world problems, we conduct extensive experiments in three MSAs with real-world data and settings. We show the information about these MSAs in Table 1. The number of vaccines available in each MSA is set in proportion to its population, i.e., 200 doses per day per 37367 people. We adopt the intrinsic parameters of COVID-19 pandemic spreading intensity in these MSAs from the original research of BD model [7], which are obtained in fitting and calibration with the real world reported situation and are shown in Appendix E.

Table 1: Details of the MSAs in experiments.

MSA	1	2	3
Name	Atlanta	Dallas	Miami
Population	7191638	8895355	6635035
Number of CBGs (N)	3130	4877	3555
Number of POIs (M)	39411	52999	40964
Number of communities (K)	7	10	8
Number of time steps (T)	1512 (Mar. 1, 12am-May 2, 11pm, 2020)		

4.2 Community Division Results

In Figure 4, we show the spatial distribution of CBGs in the divided communities and the TSNE dimension reduced POI visiting vectors of CBGs in each community, i.e., the total number of people visiting each POI over all time steps. Here we take MSA 1 as an example and the results in MSA 2 and 3 are shown in Appendix F.

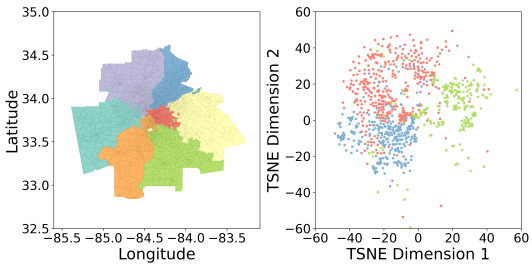


Figure 4: Community division results in MSA 1. The left panel is the spatial distribution of the CBGs and the right one is the dimension reduced POI visiting vectors of CBGs in three of the communities with corresponding color.

From the results, we verify that CBGs in the same community divided on the contact network also show good spatial locality, ensuring the real-world feasibility of vaccine allocation strategies on the community level. Besides, the POI visiting vectors of CBGs in the same community show clustering in the dimension reduced space, proving that our community division algorithm is able to capture the similarities in POI visiting patterns.

4.3 Performance Evaluation

We evaluate the performance of our method on the metrics of total cases and deaths in comparison with the following baselines:

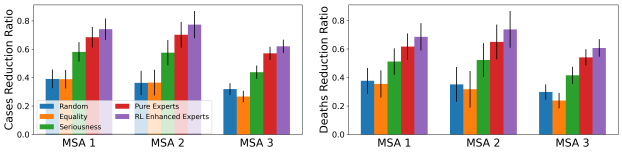
- **None**: None vaccine scenario, server as the blank control.
- **Random**: Allocating vaccines randomly among CBGs, corresponding to the situation with no specific strategy.
- **Equality** [19]: Allocating vaccines in proportion to the population of each CBG to guarantee equality.
- **Seriousness** [4]: Allocating vaccines in proportion to the total number of cases in each CBG considering the pandemic spreading seriousness.
- **Pure Experts** [21]: Allocating vaccines in proportion to new cases in the last 24 hours in each CBG as WHO experts' suggestion, considering the dynamic pandemic spreading risk. Mentioning that this baseline is identical to the expert solution layer in our RL enhanced experts method.

We train RL models in the three MSAs and obtain the testing results over 63 days with the best model. Considering the randomness in the pandemic spreading simulation, we perform 100 repeated tests and calculate the average result. We show the cases and deaths reduction comparing none vaccine scenario and the performance improvement comparing the best baseline in Table 2. We show the performance comparisons with standard deviation in Figure 5.

The results show that in all the three MSAs, our method outperforms all the baseline methods, including the pure expert solution. Quantitatively, with the same number of vaccines, our method reduces 9.01% infections and 12.27% deaths more than the pure expert

Table 2: Experimental results averaged over 100 repeated experiments. Columns 5 to 9 show the reductions relative to 'None' Scenario. The last column indicates the performance improvement of 'RL Enhanced Experts' over 'Pure Experts'.

MSA	Metric	None	Metric	Random	Equality	Seriousness	Pure Experts	RL Enhanced Experts	Performance Improvement (%)
1	Total Cases	19841.38	Cases Reduction	7758.31	7724.11	11538.14	13583.82	**14709.36**	**8.29**
	Total Deaths	662.40	Deaths Reduction	249.46	234.89	338.80	408.70	**454.09**	**11.11**
2	Total Cases	13846.38	Cases Reduction	5026.64	5059.85	7972.30	9721.15	**10703.85**	**10.11**
	Total Deaths	425.51	Deaths Reduction	149.46	134.99	222.51	276.21	**313.68**	**13.57**
3	Total Cases	27748.97	Cases Reduction	8856.13	7420.30	12166.34	15850.69	**17219.68**	**8.64**
	Total Deaths	1119.59	Deaths Reduction	333.40	266.40	465.01	605.97	**679.39**	**12.12**



(a) Cases (b) Deaths

Figure 5: Performance comparisons with standard deviation.

solution, i.e., the best baseline, on average in all the three MSAs. Actually, our method outperforms all the baselines during the whole testing process and more details are shown in Appendix G.

Besides the public health expert solutions, we also compare the performance of the following machine learning (ML) baselines:

- **GBM** [25]: A baseline for pandemic intervention by predicting the future health states, which strikes a balance between precision and recall.
- **HRLI** [15]: A state-of-the-art RL baseline for pandemic intervention by applying different strategies to people classified into different categories.

The results are shown in Table 3, which indicate that our method also outperforms the ML-based methods. Here we only take MSA 1 as an example and without loss of generality, the results in other MSAs have the same trend.

Table 3: Performance comparison with ML baselines averaged over 100 repeated experiments.

Metric	GBM	HRLI	RL Enhanced Experts
Cases Reduction	8917.76	10340.38	**14709.36**
Deaths Reduction	288.53	339.40	**454.09**

4.4 Critic Role of Contact Network and SPR

We conduct the following ablation studies to verify the critical role of multi-factor contact network and self-supervised representation:

- **No Contact Network**: We take away the whole multi-factor contact network and only keep Community States and Community Diffs vectors. We encode these two vectors with identical fully connected layers (without GCN layers) in Figure 3 and train RL models only with the PPO loss.
- **No SPR**: We keep the multi-factor contact network but take away the self-supervised representation loss by SPR in Algorithm 2 and train RL models only with the PPO loss.

We show the results of ablation studies in Table 4. From the **No Contact Network** study, we find that the performance drops sharply without the information from the multi-factor contact network, i.e., only reducing 5.44% infections and 6.78% deaths more than the pure expert solution comparing 9.01% and 12.27% with the full system. From the **No SPR** study, we find that even with the multi-factor contact network, the RL agent cannot utilize the abundant information hidden in it efficiently without SPR, and thus there is no significant performance improvement. Generally, we prove that the multi-factor contact network and the SPR method are both of vital importance, only with both of them can the RL agent obtain and well utilize adequate information about the MSAs and make efficient vaccine allocation strategies.

Table 4: Ablation study results averaged over 100 repeated experiments.

MSA	Metric	Full System	No Contact Network	No SPR
1	Cases Reduction	14709.36	14189.57	14129.48
	Deaths Reduction	454.09	431.93	431.19
2	Cases Reduction	10703.85	10128.51	10157.90
	Deaths Reduction	313.68	292.65	295.05
3	Cases Reduction	17219.68	17067.79	16842.75
	Deaths Reduction	679.39	658.82	658.05

4.5 Generalizability to More Scenarios

Previous experimental results in the three MSAs have already shown the generalizability of our method among different places. In order to test whether our method works well in more various pandemic spreading situations, we further design the following scenarios other than the original settings in Section 4.1:

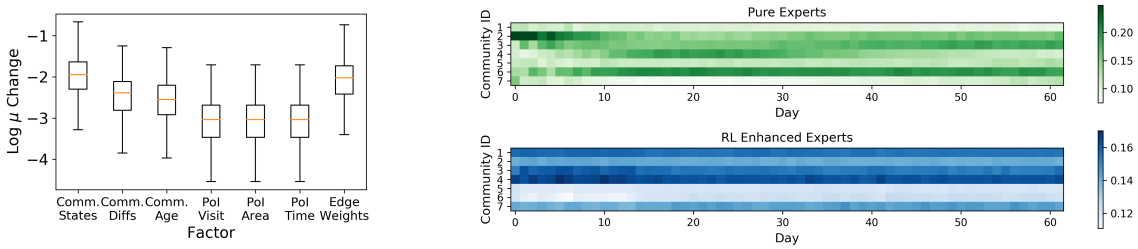
- **Scarce Vaccines**: Half number of vaccines are available.
- **Abundant Vaccines**: Doubled vaccines are available.
- **Omicron Strain**: The virus is two times more infectious, corresponding to the currently spreading Omicron strain [8].

We take MSA 1 as an example and perform 100 random tests lasting 63 days. We show the average results in Table 5.

From the results, we find that our method keeps outperforming the best baseline by a similar magnitude regardless of the number of available vaccines. Also, our method reaches higher performance gain, reducing 13.97% and 16.17% more infections and deaths when the virus's infectiousness is doubled. These results prove that our method has good generalizability and thus can be applied in either

Table 5: Experimental results on more scenarios in MSA 1 averaged over 100 repeated experiments.

Scenario	Metric	None	Metric	Random	Equality	Seriousness	Pure Experts	**RL Enhanced Experts**
Scarce Vaccines	Total Cases	19841.38	Cases Reduction	4487.14	4161.96	8762.97	10954.32	**11865.10**
	Total Deaths	662.40	Deaths Reduction	143.57	126.50	251.65	321.96	**359.92**
Abundant Vaccines	Total Cases	19841.38	Cases Reduction	11974.65	12535.53	13730.39	15321.17	**16512.47**
	Total Deaths	662.40	Deaths Reduction	391.29	390.62	412.06	469.36	**521.85**
Omicron Strain	Total Cases	316905.50	Cases Reduction	83747.85	84131.23	83544.84	100093.13	**114079.08**
	Total Deaths	12148.13	Deaths Reduction	3371.37	3201.43	3167.28	3789.89	**4402.68**



(a) Importance Analyses on Input Factors

(b) Strategies Visualization

Figure 6: (a) Importance analyses on input factors regarding the output μ of PPO actor in the last day in MSA 1. (b) Vaccine allocation strategies visualization where the color indicates the proportion of allocated vaccines.

various places or various scenarios. It is also worth mentioning that our method can be generalized with only slight adjustments to other pandemics that have similar disease models as the COVID-19.

4.6 RL Enlightens the Experts

We take MSA 1 as an example for explainability analyses with the method in Section 3.5 on the trained RL model. We show the results in Figure 6. From the importance analyses of the input factors, we find that besides the pandemic spreading situation, i.e., Community States and Community Diffs, the RL agent also pays great attention to edge weights in the contact network, which reflects the POI visiting patterns, proving the critical role of the multi-factor contact network again. This discovery enlightens the importance of population mobility and contact in vaccine allocation and pandemic intervention, which have already been adopted in some countries by tracing the contacts via wearable devices [11].

Meanwhile, the age structures of the communities, reflecting the vulnerability and death risk, are also emphasized by the RL agent. And in the strategies visualization, we notice that by considering various factors, RL agent allocates vaccines quite differently from the pure expert solution, where the latter only focuses on the pandemic spreading situation itself. Therefore, we are supposed to not only focus on the current pandemic spreading situation but also consider more about latent risks laying in the vulnerable communities, which decide long-term pandemic spreading tendency.

5 RELATED WORKS

5.1 RL for COVID-19 Intervention

The strong ability in serial decision-making of RL has long been studied to aid the COVID-19 pandemic intervention. First, RL algorithm for predicting the pandemic spreading situation to support the policymakers to optimize their policies [16] has been studied. Second, researchers have proposed various RL solutions for lockdown strategies, balancing the pandemic intervention and the side effect on social economy [17, 20].Third, an RL-based system is designed and deployed within the Greek borders for efficient and targeted COVID-19 testing [1]. Besides, RL methods for allocating medical equipment such as ventilators [2] are also proposed.

However, it is almost unexplored how to allocate COVID-19 vaccines, especially in the large-scale metropolis scenario. We are the first to solve this problem in our work.

5.2 Self-supervised Representation Learning

Self-supervised representation learning aims to obtain efficient feature representations for downstream tasks by introducing some unsupervised auxiliary tasks. Recently, it has been widely used in the fields of natural language processing [9], computer vision [14] and reinforcement learning [29, 30, 32, 34, 35].

In RL-related works, SAC-AE [34] introduces auxiliary tasks such as pixel-level reconstruction. CURL [30] applies image augmentation to generate positive and negative pairs and uses them for contrastive learning. However, these representation learning methods for image observations are difficult to be applied to other types of inputs. Further, ATC [32] avoids such a problem by generating positive and negative pairs and contrastive losses according to the temporal relationship. SPR [29] and PlayVirtual [35] further introduce dynamics modeling to improve data efficiency.

Generally, existing works mainly focus on representing the input of images while it is almost unexplored how to deal with graph-structured input, which is commonly seen in many scenarios though. In contrast, we focus on representing the multi-factor contact network to support the following RL task in this paper.

6 CONCLUSIONS

In this paper, we studied the problem of large-scale COVID-19 vaccine allocation. We proposed an RL enhanced experts method, which efficiently utilizes multi-factor contact network information via self-supervised representation learning. Extensive experiments in three MSAs under various scenarios with real-world data prove that our method outperforms all the baselines and has good generalizability in solving real-world problems. Meanwhile, the explainability analyses on the trained RL model not only add to the credibility and feasibility of our method in the real world but also provide enlightenment to the experts and benefit future decision-making.

By doing these, our work not only provides technical innovations but is also applicable in addressing real-world challenges. Experimental results show that our method is especially efficient in scenarios with a more infectious virus, which is helpful in dealing with the currently spreading Omicron strain. The generalizability indicates that our method has the potential to be applied in more scenarios and more places around the world, especially in undeveloped regions where vaccines are in shortage. Furthermore, our method can be adapted to other pandemics with only slight adjustments. All these advantages contribute to advancing global good health and well-being greatly.

ACKNOWLEDGMENTS

This work was supported in part by The National Key Research and Development Program of China under grant 2020AAA0106000, The National Natural Science Foundation of China under U21B2036, U20B2060, 61971267.

REFERENCES

[1] Hamsa Bastani, Kimon Drakopoulos, Vishal Gupta, Ioannis Vlachogiannis, Christos Hadjicristodoulou, Pagona Lagiou, Gkikas Magiorkinis, Dimitrios Paraskevis, and Sotirios Tsiodras. 2021. Efficient and targeted COVID-19 border testing via reinforcement learning. *Nature* 599, 7883 (2021), 108–113.

[2] Bryan P Bednarski, Akash Deep Singh, and William M Jones. 2021. On collaborative reinforcement learning to optimize the redistribution of critical medical supplies throughout the COVID-19 pandemic. *Journal of the American Medical Informatics Association* 28, 4 (2021), 874–878.

[3] Kate M Bubar, Kyle Reinholt, Stephen M Kissler, Marc Lipsitch, Sarah Cobey, Yonatan H Grad, and Daniel B Larremore. 2021. Model-informed COVID-19 vaccine prioritization strategies by age and serostatus. *Science* 371, 6532 (2021), 916–921.

[4] Hui Cao and Simin Huang. 2012. Principles of scarce medical resource allocation in natural disaster relief: a simulation approach. *Medical Decision Making* 32, 3 (2012), 470–476.

[5] Serina Chang, Emma Pierson, Pang Wei Koh, Jaline Gerardin, Beth Redbird, David Grusky, and Jure Leskovec. 2021. Mobility network models of COVID-19 explain inequities and inform reopening. *Nature* 589, 7840 (2021), 82–87.

[6] Serina Chang, Mandy L Wilson, Bryan Lewis, Zakaria Mehrab, Komal K Dudakiya, Emma Pierson, Pang Wei Koh, Jaline Gerardin, Beth Redbird, David Grusky, et al. [n.d.]. Supporting covid-19 policy response with large-scale mobility-based modeling. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*.

[7] Lin Chen, Fengli Xu, Zhenyu Han, Kun Tang, Pan Hui, James Evans, and Yong Li. 2021. Strategic COVID-19 vaccine distribution can simultaneously elevate social utility and equity. *arXiv preprint arXiv:2111.06689* (2021).

[8] Carlos Del Rio, Saad B Omer, and Preeti N Malani. 2021. Winter of omicron—the evolving COVID-19 pandemic. *JAMA* (2021).

[9] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805* (2018).

[10] Ezekiel J Emanuel, Govind Persad, Adam Kern, Allen Buchanan, Cécile Fabre, Daniel Halliday, Joseph Heath, Lisa Herzog, RJ Leland, Ephrem T Lemango, et al. 2020. An ethical framework for global vaccine allocation. *Science* 369, 6509 (2020), 1309–1312.

[11] Jim AC Everett, Clara Colombatto, Edmond Awad, Paulo Boggio, Björn Bos, William J Brady, Megha Chawla, Vladimir Chituc, Dongil Chung, Moritz A Drupp, et al. 2021. Moral dilemmas and trust in leaders during a global health crisis. *Nature human behaviour* 5, 8 (2021), 1074–1088.

[12] Samuel Greydanus, Anurag Koul, Jonathan Dodge, and Alan Fern. 2018. Visualizing and understanding atari agents. In *International conference on machine learning*. PMLR, 1792–1801.

[13] Qianyue Hao, Lin Chen, Fengli Xu, and Yong Li. 2020. Understanding the urban pandemic spreading of COVID-19 with real world mobility data. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 3485–3492.

[14] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. 2020. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*.

[15] Yuanshuang Jiang, Linfang Hou, Yuxiang Liu, Zhuoye Ding, Yong Zhang, and Shengzhong Feng. 2020. Epidemic Control Based on Reinforcement Learning Approaches. (2020).

[16] Soheyl Khalilpourazari and Hossein Hashemi Doulabi. 2021. Designing a hybrid reinforcement learning based algorithm with application in prediction of the COVID-19 pandemic in Quebec. *Annals of Operations Research* (2021), 1–45.

[17] Gloria Hyunjung Kwak, Lowell Ling, and Pan Hui. 2021. Deep reinforcement learning approaches for global public health strategies for COVID-19 pandemic. *Plos one* 16, 5 (2021), e0251550.

[18] Laura Matrajt, Julia Eaton, Tiffany Leung, and Elizabeth R Brown. 2021. Vaccine optimization for COVID-19: Who to vaccinate first? *Science Advances* 7, 6 (2021), eabf1374.

[19] John N Nkengasong, Nicaise Ndembi, Akhona Tshangela, and Tajudeen Raji. 2020. COVID-19 vaccines: how to ensure Africa has access.

[20] Abu Quwsar Ohi, MF Mridha, Muhammad Mostafa Monowar, Md Hamid, et al. 2020. Exploring optimal control of epidemic spread using reinforcement learning. *Scientific reports* 10, 1 (2020), 1–19.

[21] World Health Organization et al. 2020. *WHO SAGE values framework for the allocation and prioritization of COVID-19 vaccination, 14 September 2020*. Technical Report. World Health Organization.

[22] Ferran Parés, Dario Garcia Gasulla, Armand Vilalta, Jonatan Moreno, Eduard Ayguadé, Jesús Labarta, Ulises Cortés, and Toyotaro Suzumura. 2017. Fluid communities: a competitive, scalable and diverse community detection algorithm. In *International conference on complex networks and their applications*. Springer, 229–240.

[23] Govind Persad, Ezekiel J Emanuel, Samantha Sangenito, Aaron Glickman, Steven Phillips, and Emily A Largent. 2021. Public perspectives on COVID-19 vaccine prioritization. *JAMA network open* 4, 4 (2021), e217943–e217943.

[24] Govind Persad, Monica E Peek, and Ezekiel J Emanuel. 2020. Fairly prioritizing groups for access to COVID-19 vaccines. *Jama* 324, 16 (2020), 1601–1602.

[25] Stefano Giovanni Rizzo. 2020. Balancing precision and recall for costeffective epidemic containment. (2020).

[26] Harald Schmidt, Rebecca Weintraub, Michelle A Williams, Kate Miller, Alison Buttenheim, Emily Sadecki, Helen Wu, Aditi Doiphode, Neha Nagpal, Lawrence O Gostin, et al. 2021. Equitable allocation of COVID-19 vaccines in the United States. *Nature Medicine* 27, 7 (2021), 1298–1307.

[27] John Schulman, Philipp Moritz, Sergey Levine, Michael Jordan, and Pieter Abbeel. 2015. High-dimensional continuous control using generalized advantage estimation. *arXiv preprint arXiv:1506.02438* (2015).

[28] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347* (2017).

[29] Max Schwarzer, Ankesh Anand, Rishab Goel, R Devon Hjelm, Aaron Courville, and Philip Bachman. 2020. Data-efficient reinforcement learning with self-predictive representations. *arXiv preprint arXiv:2007.05929* (2020).

[30] Aravind Srinivas, Michael Laskin, and Pieter Abbeel. 2020. Curl: Contrastive unsupervised representations for reinforcement learning. *arXiv preprint arXiv:2004.04136* (2020).

[31] Andrew J Stier, Marc G Berman, and Luis Bettencourt. 2020. COVID-19 attack rate increases with city size. *arXiv preprint arXiv:2003.10376* (2020).

[32] Adam Stooke, Kimin Lee, Pieter Abbeel, and Michael Laskin. 2021. Decoupling representation learning from reinforcement learning. In *International Conference on Machine Learning*. PMLR, 9870–9879.

[33] Bing Xu, Naiyan Wang, Tianqi Chen, and Mu Li. 2015. Empirical evaluation of rectified activations in convolutional network. *arXiv preprint arXiv:1505.00853* (2015).

[34] Denis Yarats, Amy Zhang, Ilya Kostrikov, Brandon Amos, Joelle Pineau, and Rob Fergus. 2019. Improving sample efficiency in model-free reinforcement learning from images. *arXiv preprint arXiv:1910.01741* (2019).

[35] Tao Yu, Cuiling Lan, Wenjun Zeng, Mingxiao Feng, Zhizheng Zhang, and Zhibo Chen. 2021. Playvirtual: Augmenting cycle-consistent virtual trajectories for reinforcement learning. *Advances in Neural Information Processing Systems* 34 (2021).

A IMPLEMENTATION DETAILS FOR REPRODUCIBILITY

We perform training and testing using Python 3.8 and Pytorch 1.10 with NVIDIA GeForce RTX 3090 GPUs. Here we provide detailed values of the hyper-parameters in Table 6 for reproducibility.

Table 6: Values of the hyper-parameters.

Hyper-parameter	Notation in the context	Value
Encounter threshold	ξ	10
Average community size	S	500
Small item in normalization	ϵ_0	1×10^{-5}
Clip bounds for Community Age	(δ_l, δ_u)	(0,1)
Clip bounds for POI Areas	(δ_l, δ_u)	(0,0.99)
Clip bounds for POI Times	(δ_l, δ_u)	(0,0.99)
Clip bounds for edge weights	(δ_l, δ_u)	(0,0.99)
Slope of the Leaky-ReLU function	–	0.01
EMA ratio	κ	0.5
Batch size (MSA 1 and 2)	N_B	256
Batch size (MSA 3)	N_B	128
Training epoch	E	10
Replay buffer size	B	8192
Weight for SPR loss	λ	2
Optimizer	–	Adam
Learning rate	–	1×10^{-4}
Small item in reward function	ϵ_r	1
Long term reward decay rate	γ	0.6
Small item in PPO clip bound	ϵ	0.2

B VECTOR NORMALIZATIONS IN SPR

In this section, we show the details of the three different vector normalization methods in the neural network for different types of data. For dynamic feature vectors, we perform typical batch normalization in the deep neural network as follows:

$$y = \frac{x - Mean[x]}{\sqrt{Var[x] + \epsilon_0}} \times \gamma + \beta, \tag{12}$$

where mean and variance are calculated over the mini-batches and γ, β are learnable affine parameters. For static feature vectors, we perform the following normalization:

$$y = \frac{x - Mean[Clip(x, \delta_l x, \delta_u x)]}{\sqrt{Var[Clip(x, \delta_l x, \delta_u x)]}}, \tag{13}$$

where $Clip(x, a, b)$ equals to $Min[Max[x, a], b]$. For edge weights, which are all positive numbers, we simply use:

$$y = \frac{x}{Max[Clip(x, \delta_l x, \delta_u x)]}. \tag{14}$$

where maximum is calculated over the mini-batches. The hyper-parameters $0 \leq \delta_l < \delta_u \leq 1$ in the latter twos constrain the lower and upper bounds of data in calculating the mean value, standard-deviation or maximum value, avoiding the situation that most of the elements are over-suppressed by several extreme ones, which occurs when the data exhibits a long-tailed distribution.

C NEURAL NETWORK STRUCTURES IN SPR

In this section, the detailed structures of the auxiliary neural networks in the SPR training for self-supervised representation learning are shown as follows:

- **State transition model**: This neural network takes in the current state z_t and action a_t and predicts the next step state $\hat{z}_{t+1}$. The current state head consists of two fully connected layers FC [16, 32] and the action head consists of two fully connected layers FC [8, 16]. The outputs of the two heads are concatenated and then go through two fully connected layers FC [16, 8] to reach the final output.
- **Projection**: This neural network project the state embeddings into lower dimensional space for SPR loss calculation. First, the input tensor is reshaped into a one-dimensional vector and then it goes through three fully connected layers FC [128, 64, 16] to reach the final output.
- **Prediction**: This is the additional neural network on the online branch before SPR loss calculation. It consists of three fully connected layers FC [32, 32, 16].

All activation functions between layers in the above neural networks are Leaky-ReLU functions.

D DETAILS IN THE RL TRAINING

In this section, we show more details in the RL training process which are omitted in Section 3.4. First, detailed structures of the actor and critic neural network are as follows:

- **Actor**: This neural network takes in the state embeddings and outputs μ and σ for action sampling from a multivariate normal distribution. First, the tensor of the state embeddings is reshaped into a one-dimensional vector and then passes through a backbone network with two fully connected layers FC [128, 64]. Then for the μ branch, the backbone outputs pass through two fully connected layers FC [64, K], and for the σ branch, the backbone outputs also pass through two fully connected layers FC [64, K], where K is the number of communities.
- **Critic**: This neural network takes in the state embeddings and outputs the estimation of the state value. The tensor of the state embeddings is reshaped into a one-dimensional vector and then passes through four fully connected layers FC [128, 64, 16, 1] to reach the final output.

Second, the detailed mathematical formulation of the **SmoothL1** loss function we used in the calculation of the critic loss l_c is:

$$\textbf{SmoothL1}(x, y) = \frac{1}{n}\Sigma_i z_i, \tag{15}$$

where

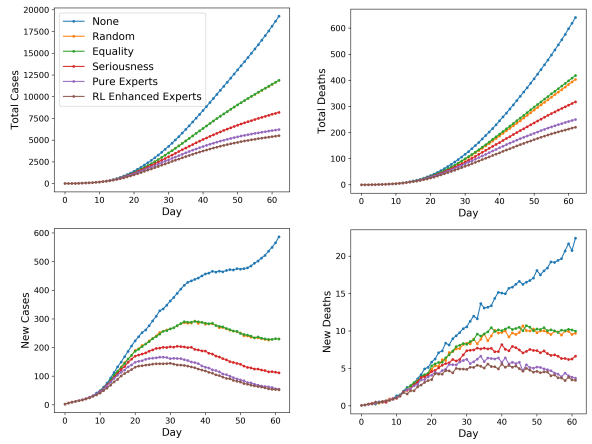
$$z_i = \begin{cases} 0.5(x_i - y_i)^2, \mid x_i - y_i \mid < 1 \\ \mid x_i - y_i \mid -0.5, otherwise. \end{cases} \tag{16}$$

Third, we train RL models in all the three MSAs for 30 episodes (1512 steps, i.e., 63 days, each episode) with 128 parallel computed different random seeds.

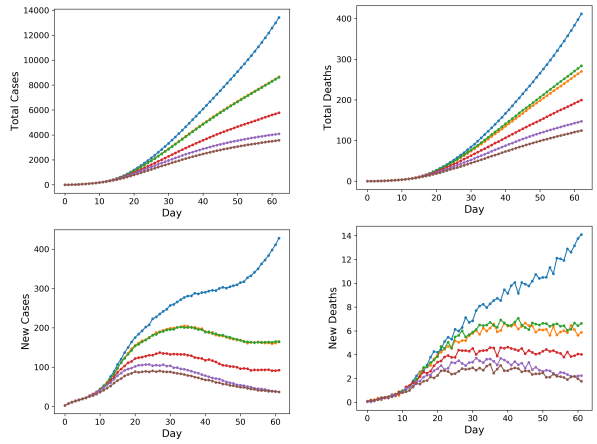
E INTRINSIC PARAMETERS OF COVID-19

In this section, we show the intrinsic parameters describing the different intensity of COVID-19 pandemic spreading in the three experimental MSAs in fitting and calibration with the real world reported situation, including:

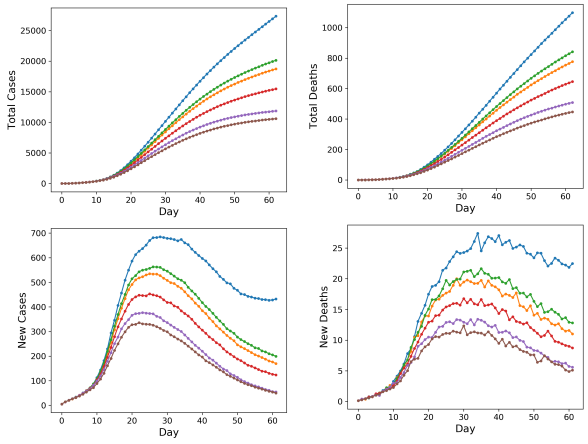
- p_0: Proportion of initially infected cases at the start step.
- β: Scaling factor on the probability of COVID-19 transmission happened in CBGs.



(a) MSA 1



(b) MSA 2



(c) MSA 3

Figure 8: Curves of cases and deaths during the whole testing process with different strategies.

- ψ: Scaling factor on the probability of COVID-19 transmission happened in POIs.
- S_d: Scaling factor on the death rate.

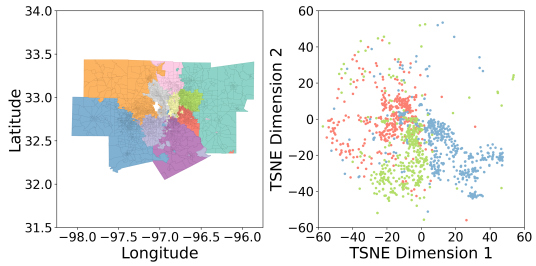
We show the exact value of p_0 and the dimensionless relative values of the latter three scaling factors in Table 7 for intuitive understanding. We adopt their absolute values in our implementation from the BD model paper [7].

Table 7: Intrinsic parameters of COVID-19.

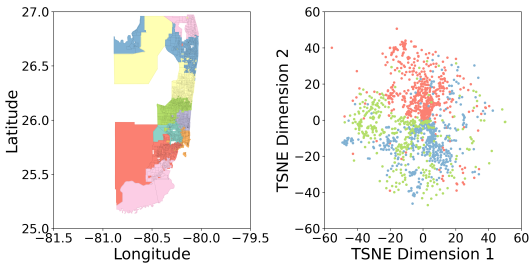
Parameter	MSA 1	MSA 2	MSA 3
p_0	2×10^{-4}	2×10^{-4}	5×10^{-4}
β	0.59	1.00	0.19
ψ	1.00	0.61	0.74
S_d	1.00	0.86	0.65

F COMMUNITY DIVISION RESULTS

In Figure 7, we show the spatial distribution of CBGs in the divided communities and the TSNE dimension reduced POI visiting vectors of CBGs in each community, in MSA 2 and 3.



(a) MSA 2



(b) MSA 3

Figure 7: Community division results in MSA 2 and 3. The left panels are the spatial distribution of the CBGs and the right ones are the dimension reduced POI visiting vectors of CBGs in three of the communities with corresponding color.

G DETAILED EXPERIMENTAL RESULTS DURING THE WHOLE TESTING PROCESS

We show the detailed curves of cases and deaths during the whole testing process with different strategies in Figure 8. From the results, we prove that our method not only reduces the final total cases and deaths but also outperforms all the baseline methods during the whole testing process.