Reversed Plague Inc.

”Simulation of Spreading And Containment of a Virus by Means of AI Techniques and Planning”

Cognolato Samuel

Fusaro Daniel

Trombin Edoardo

Index

Problem Statement

The diffusion of **CoVid-19** has suddenly changed the way people lives every day.

The current situation has shown that the right combination of containment measures can dramatically reduce the spreading of the virus, and the consequent human and economic losses.

Reversed Plague Inc. is an application designed to simulate the spreading of a Virus in an Urban Environment.

An Agent can enact or disable different containment measures during the simulation.

Agent's goal:

End the spreading of the virus:

- in the shortest possible **time**,
- With lowest number of **deaths**
- With lowest **economic** loss

through the use of a **reinforcement learning** algorithm.

SIR Model

Susceptible

↓

Infected

↓

Recovered

$$\frac{dS}{dt} = -\frac{\beta I S}{N}$$

$$\frac{dI}{dt} = \frac{\beta I S}{N} - \gamma I$$

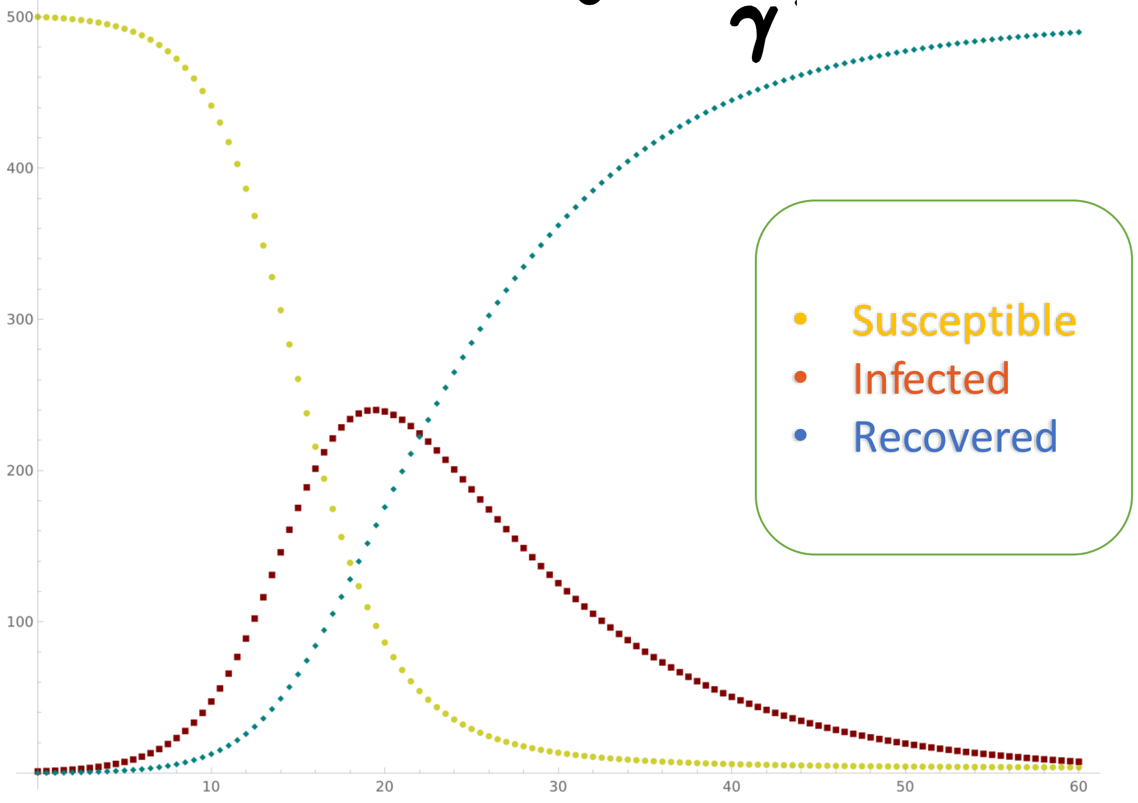
$$\frac{dR}{dt} = \gamma I$$

Equilibrium Equation

$$\frac{dS}{dt} + \frac{dI}{dt} + \frac{dR}{dt} = 0$$

Basic Reproduction Number

$$R_0 = \frac{\beta}{\gamma}$$

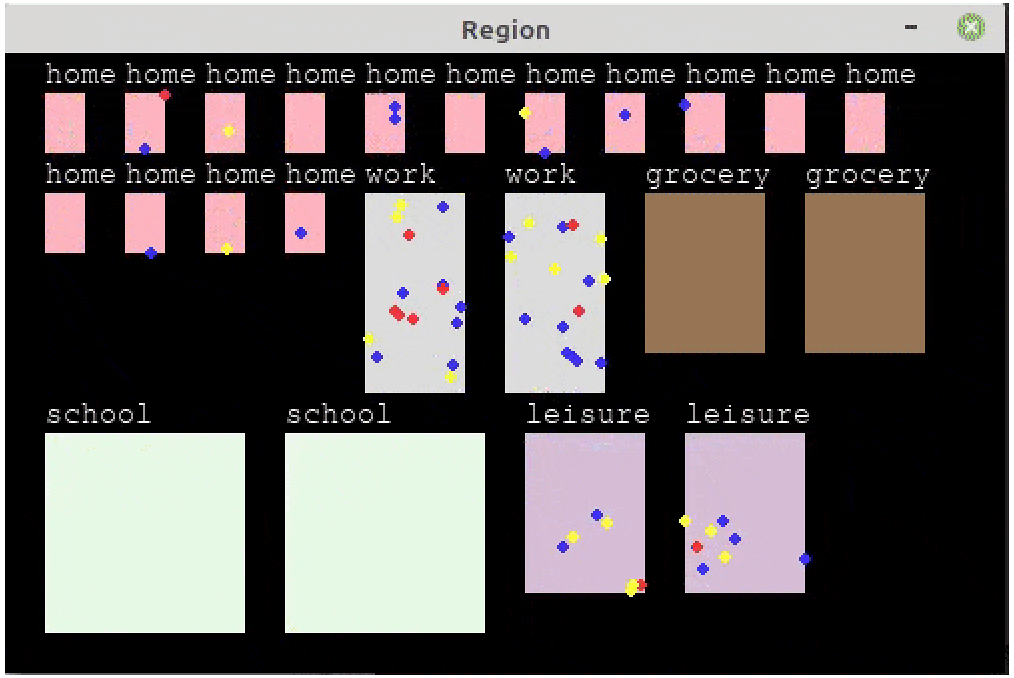


Simulation Environment

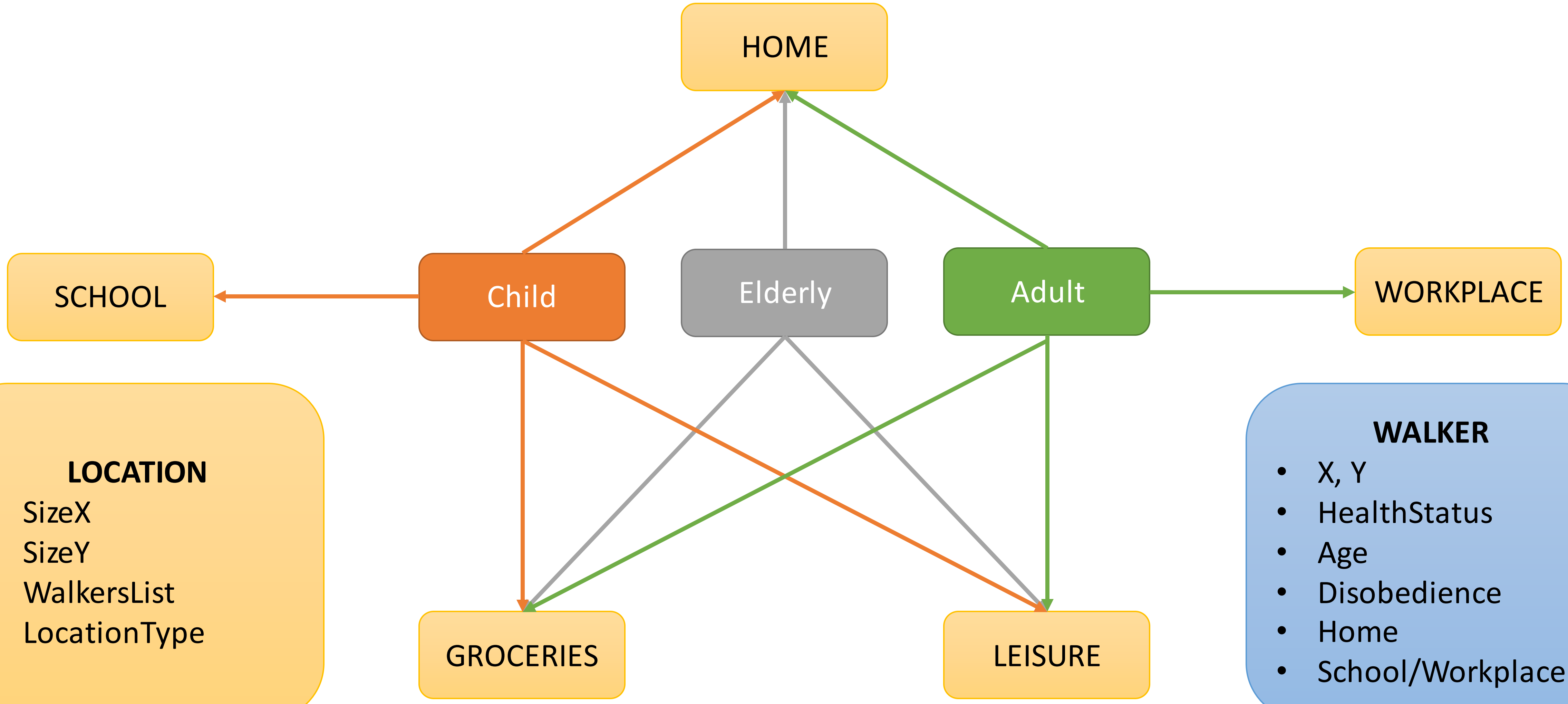
It models a **City**, where a person (Walker) accesses **various locations** based on its **schedule**.

Task Environment Properties:

- Partially Observable
- Stochastic
- Sequential
- Dynamic
- Discrete
- Single-Agent



Locations and Walkers



LOCATION

- SizeX
- SizeY
- WalkersList
- LocationType

WALKER

- X, Y
- HealthStatus
- Age
- Disobedience
- Home
- School/Workplace

Virus Propagation Model

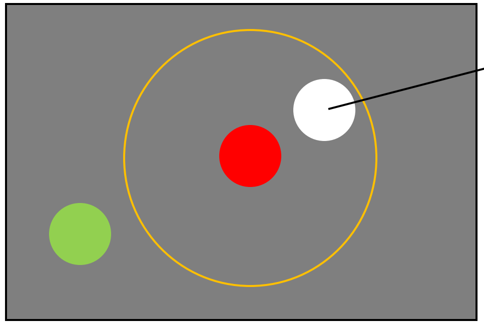
The virus gets the chance to **spread** when an infected walker gets close to a healthy walker.

After the virus incubation time passes, the walker will reach one of the following states:

- **Infected:** the walker gets sick and can spread the infection further;
- **Asymptomatic:** the walker remains virtually healthy but can spread the virus regardless.

After the illness period, the walker will reach one of the following states:

- **Recovered:** the walker becomes healthy again and is now immune to the virus;
- **Dead:** the walker disappears from the simulation.



A susceptible walker have a certain probability of getting infected

Statistics and Virus Containment

Statistics are computed at the end of every day of the simulation, and are the core of the **agent's observation space** (the set of values observed to evaluate the "goodness" of a choice).

These statistics include:

- **S,I,R:** number of Susceptible, Infected, Recovered, and Dead individuals;
- **R_0:** virus' basic reproduction number;
- **M:** average amount of money possessed by families;
- **D:** discontent of the population, influenced by the current state of the environment and the agent's choices.

Agent : Action Space

The agent goal is to choose the best combination of **containment measures** to contain the diffusion of the virus.

They can be enabled and disabled, and each measure alters different components of the simulation environment, lowering some virus-related parameters while increasing population's discontent.

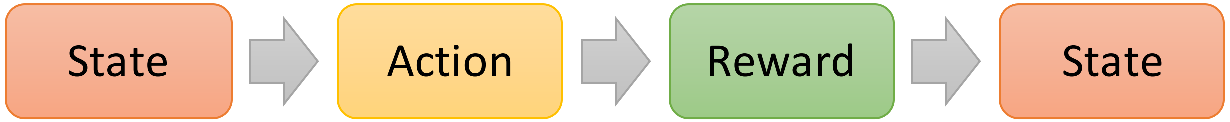
The measures include:

- **Mandatory Protective Mask:** influences infection probability;
- **Mandatory Social Distancing:** influences walkers' movements;
- **Quarantine Mode:** influences walker's movement by confining the infected ones;
- **Location Closing:** influences the accessibility of some type of locations (e.g. Closing all schools).

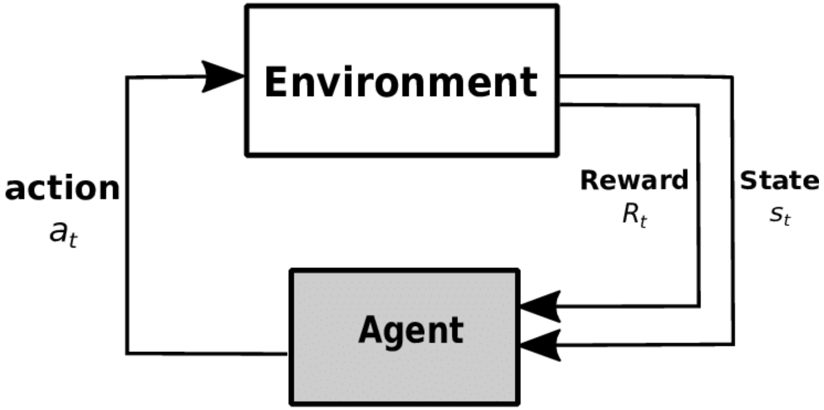
Reinforcement Learning: an Introduction

It is a general approach for which an agent learns to take actions in an environment in order to maximize the reward.

Q-learning is a particular reinforcement learning algorithm which involves a <u>sequence of steps</u> and a <u>quality function</u> to estimate.



State – action quality function: $\boldsymbol{Q : S \times A \to \mathbb{R}}$



Function update:

$$Q(s_t,a_t) \leftarrow \underbrace{Q(s_t,a_t)}_{\text{vecchio valore}} + \underbrace{\alpha_t(s_t,a_t)}_{\text{tasso di apprendimento}} \times \left[\overbrace{\underbrace{r_t}_{\text{ricompensa}} + \underbrace{\gamma}_{\text{fattore di sconto}} \underbrace{\max_{a_{t+1}} Q(s_{t+1},a_{t+1})}_{\text{valore futuro massimo}}}^{\text{valore appreso}} - \underbrace{Q(s_t,a_t)}_{\text{vecchio valore}} \right]$$

A focus on: reward evaluation

The reward is calculated as the weighted sum of functions of each parameter in the observation space.

The agent will learn which action gives the most benefit in the current state considering a trade-off between the immediate and the delayed reward.

Depends on:

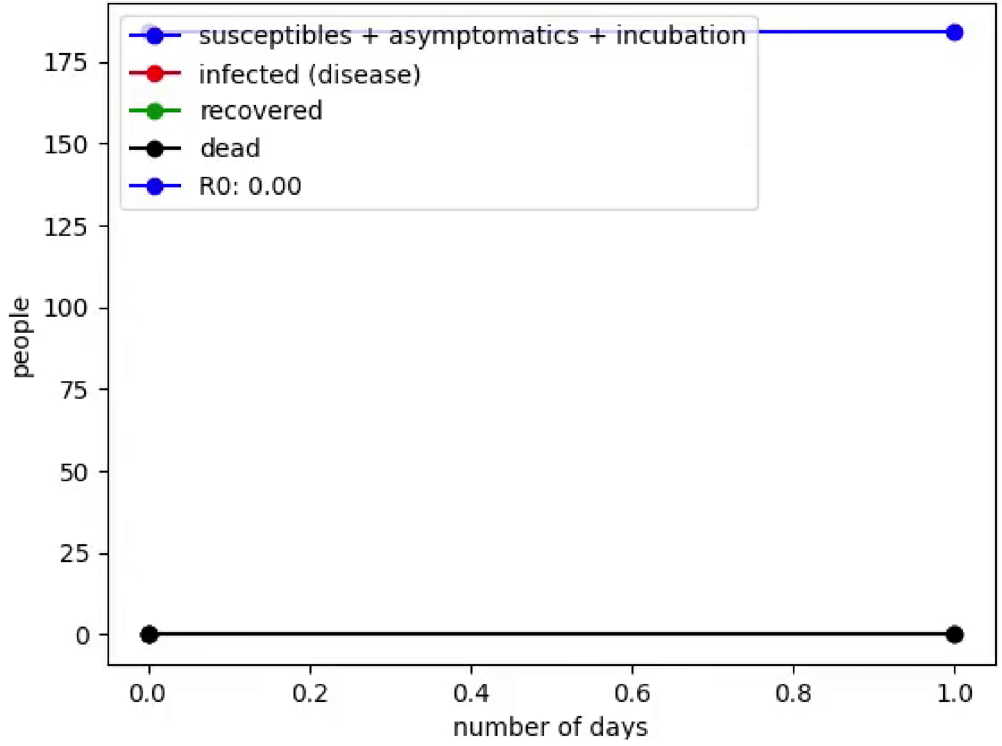
- Impossibility to go to leisure places;
- Food availability;
- Unpleasant containment measures.

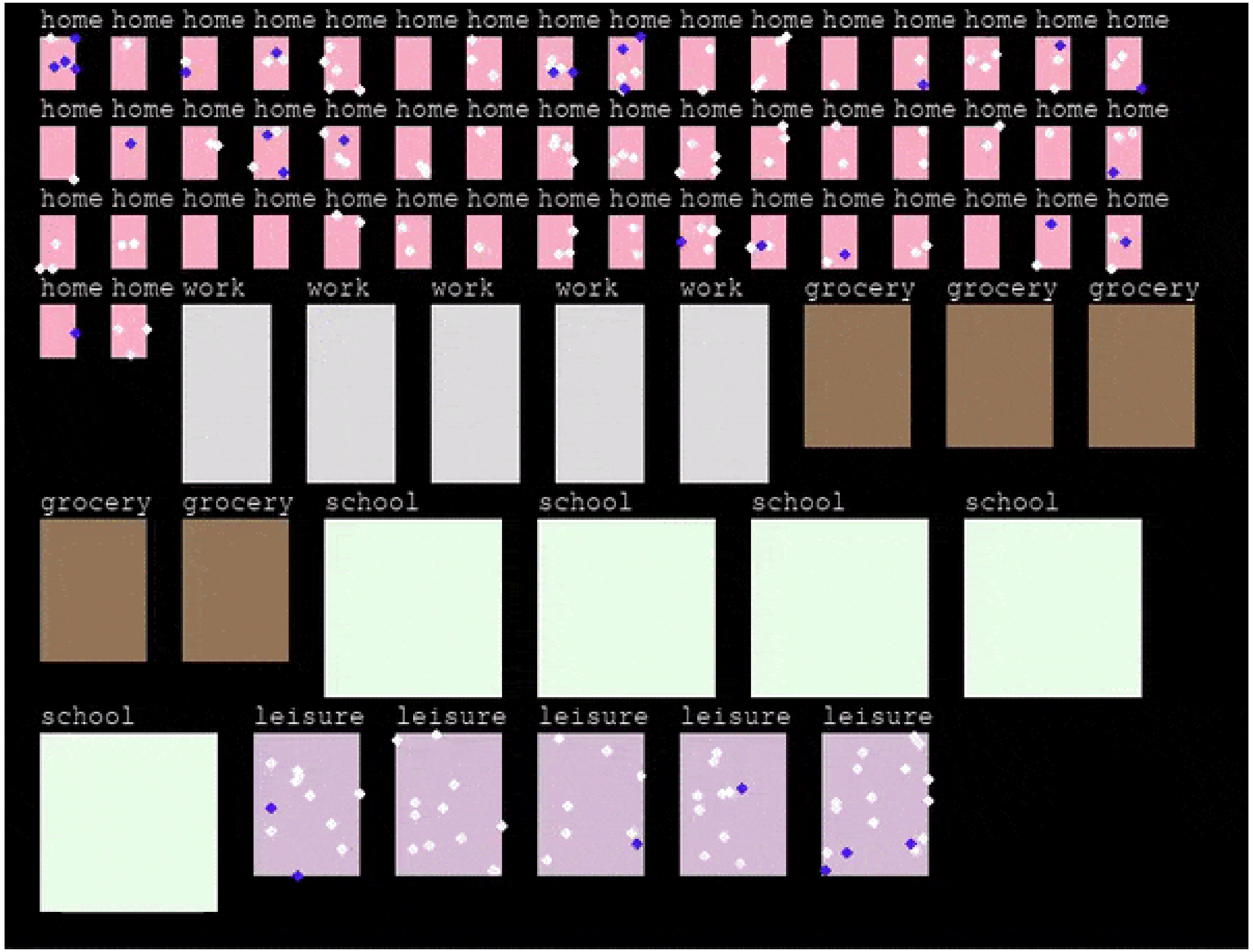
Reward:

- increased with an $R_0<1$
- Decreased with an $R_0>1$

$$Reward = -\alpha \cdot (Discontent) + \beta \cdot (\frac{1}{R_0} - R_0) - \gamma\,(Deaths) + \delta\,(MoneyAvg)$$

State of The Project





Further Improvements

This work we did is very modular and extendable. Future improvements may regard:

- The possibility to **swab** walkers to find the infected (also among the asymptomatics);
- New locations such as **hospitals** to allow for the recovery of walkers & stress test on the medical system;
- Extension on the regions, to provide comunication between **cities**;
- Refine the **effects** of the containment measures;
- **Efficiency** improvement.

Thank You!

External References

GitHub repository of the project

https://github.com/Bender97/RPlagueInc

SIR Model

https://arxiv.org/pdf/2003.14098.pdf

OpenAI/Gym

https://github.com/openai/gym