

A TypeScript Based Agent-based Disease Mode

A Disease Model in TypeScript

Heyan Zhu (Anson) Mathematics - Computer Science Class

Introduction

The background of Disease Prediction Models

Introduction

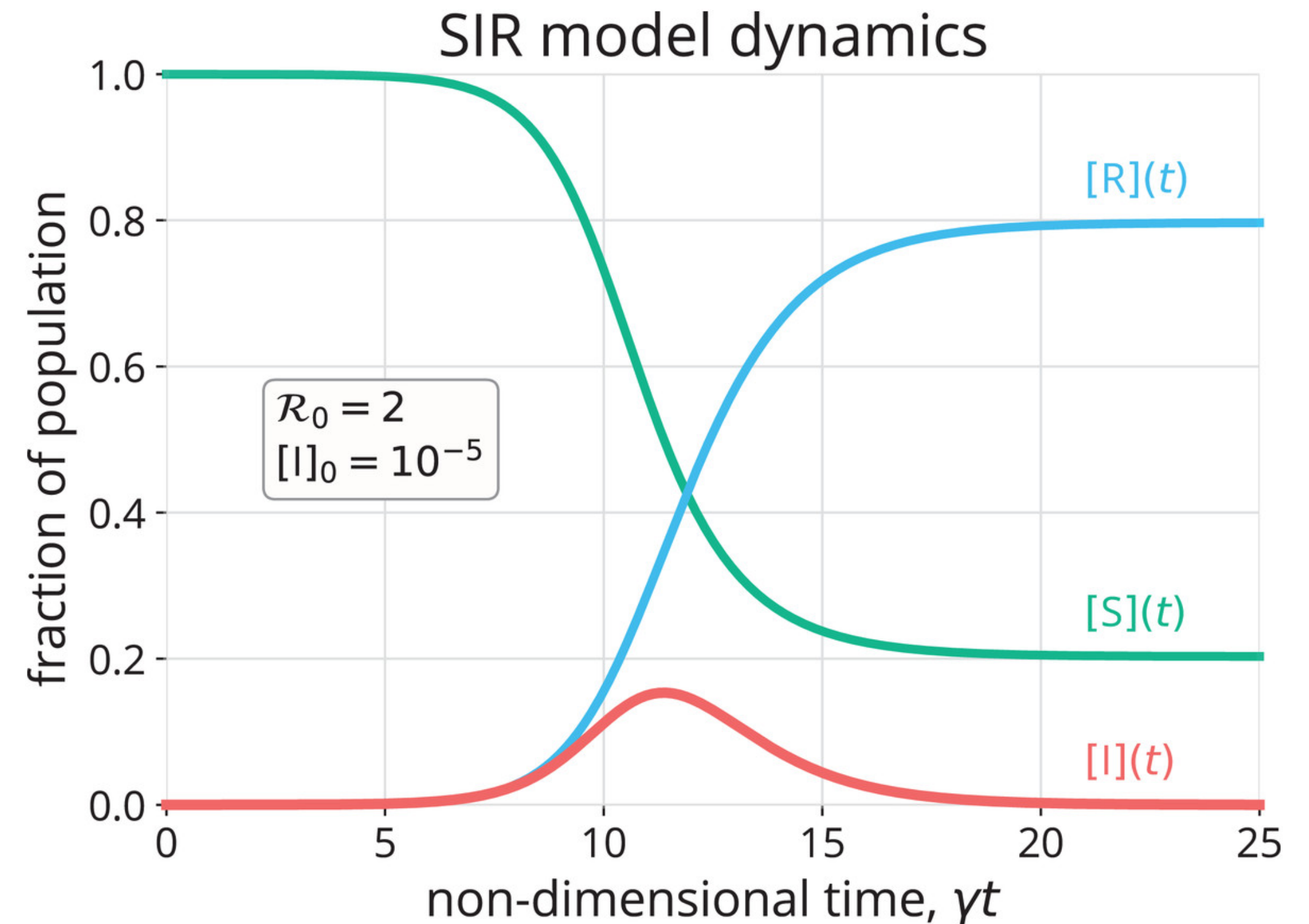
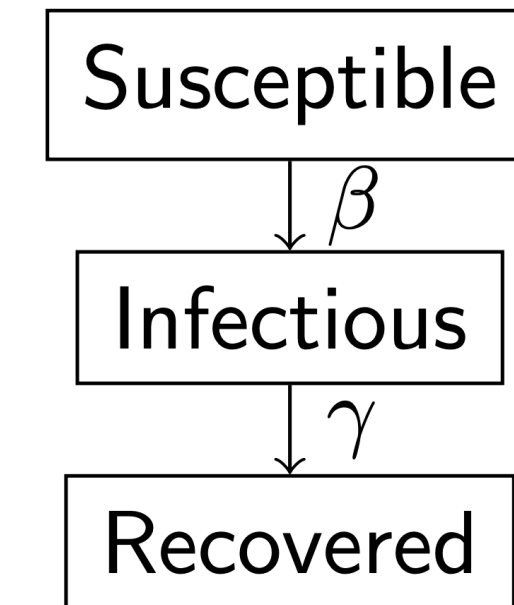
The background of Disease Prediction Models

- **SIR model**
- A set of **differential** equations
- Describes the number of people
 - **Susceptible** (S, not infected)
 - **Infected** (I)
 - **Recovered** (R, recovered, not susceptible)

$$\frac{dS}{dt} = -\beta SI$$

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

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

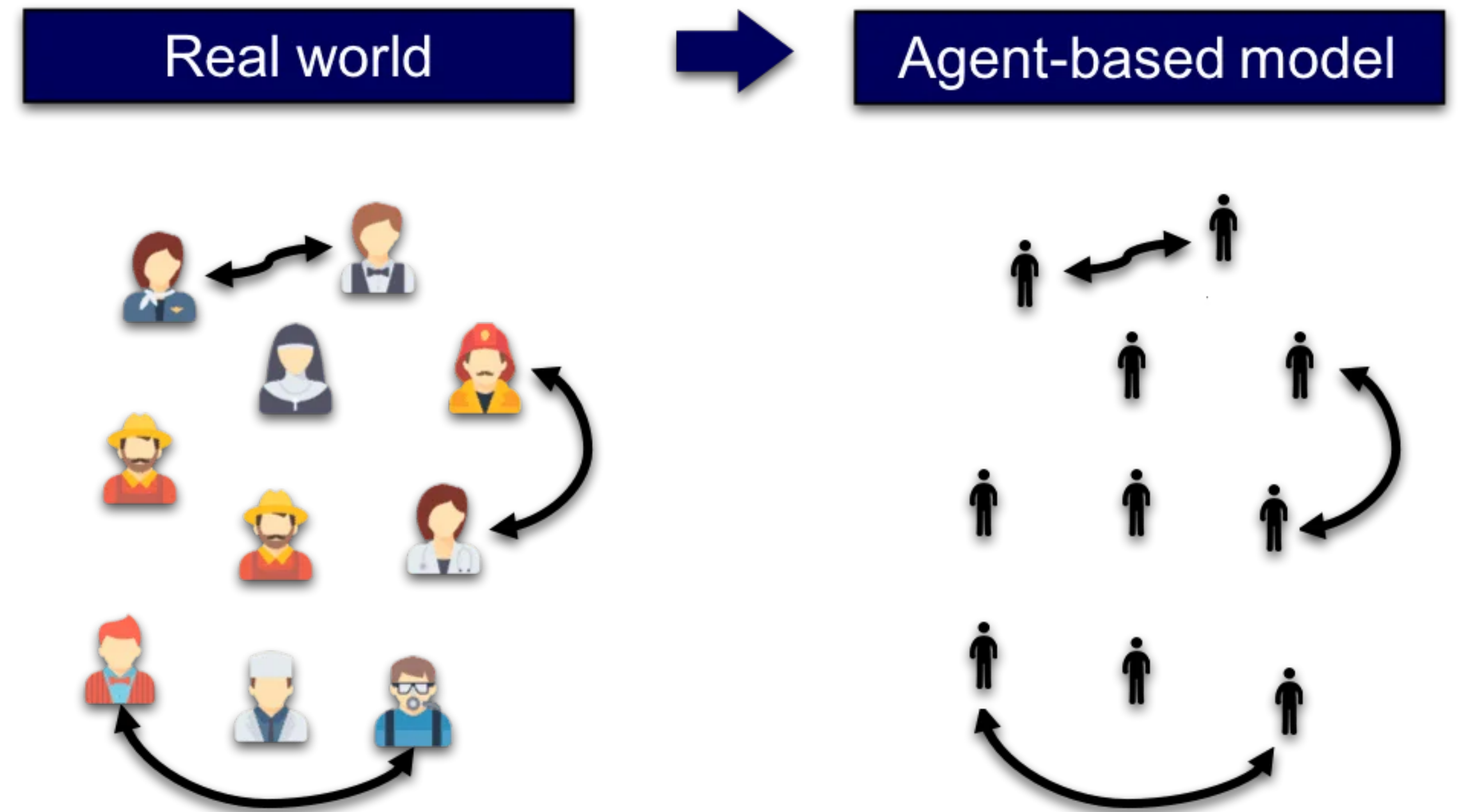


This image comes from: <https://peerj.com/articles/pchem-14/>

Introduction

The background of Disease Prediction Models

- **Agent-based model**
 - A computer **simulation** of the **interaction** of agents (people)
 - Used to **predict** and **demonstrate** the spread of diseases
- Agent is associated with certain attributes



This image comes from: <https://www.synlabs.io/post/top-5-significant-use-cases-of-agent-based-modeling-simulation>

Methodology

Constructing the Agent-based Model

Agent-based Model

Building blocks

- **Components:**
 - **Agents:** Collections of **properties** (name, age, gender, health, etc.)
 - **Behaviours:** Daily **routines** and **interactions**.
 - **Environment:** Space where agents **reside and interact**.

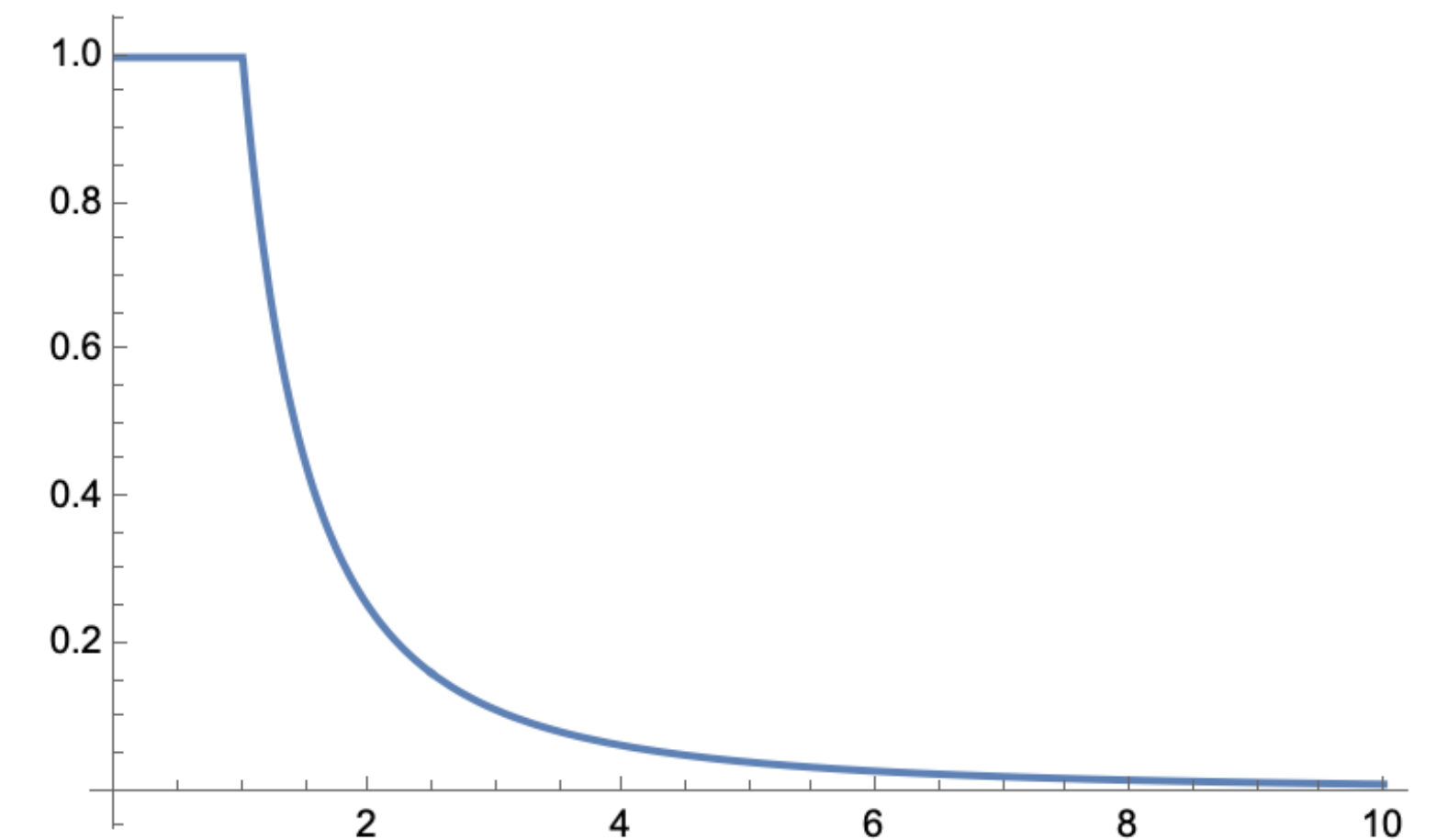
Machinist of Infection

The process that the disease spread

- The **risk of infection** will be **increased** as:
 - The susceptible agents **moved close** to the **infected** agents.

$$\mathbb{P}(\text{Infection}) = \begin{cases} \frac{1}{d^2} & (\text{for } d > 1) \\ 1 & (\text{for } d \leq 1) \end{cases} \quad (d \text{ is the between the agents})$$

- A **timer** will start if the agents get infected.
- Agents will **recover** if the timer **reaches zero**.

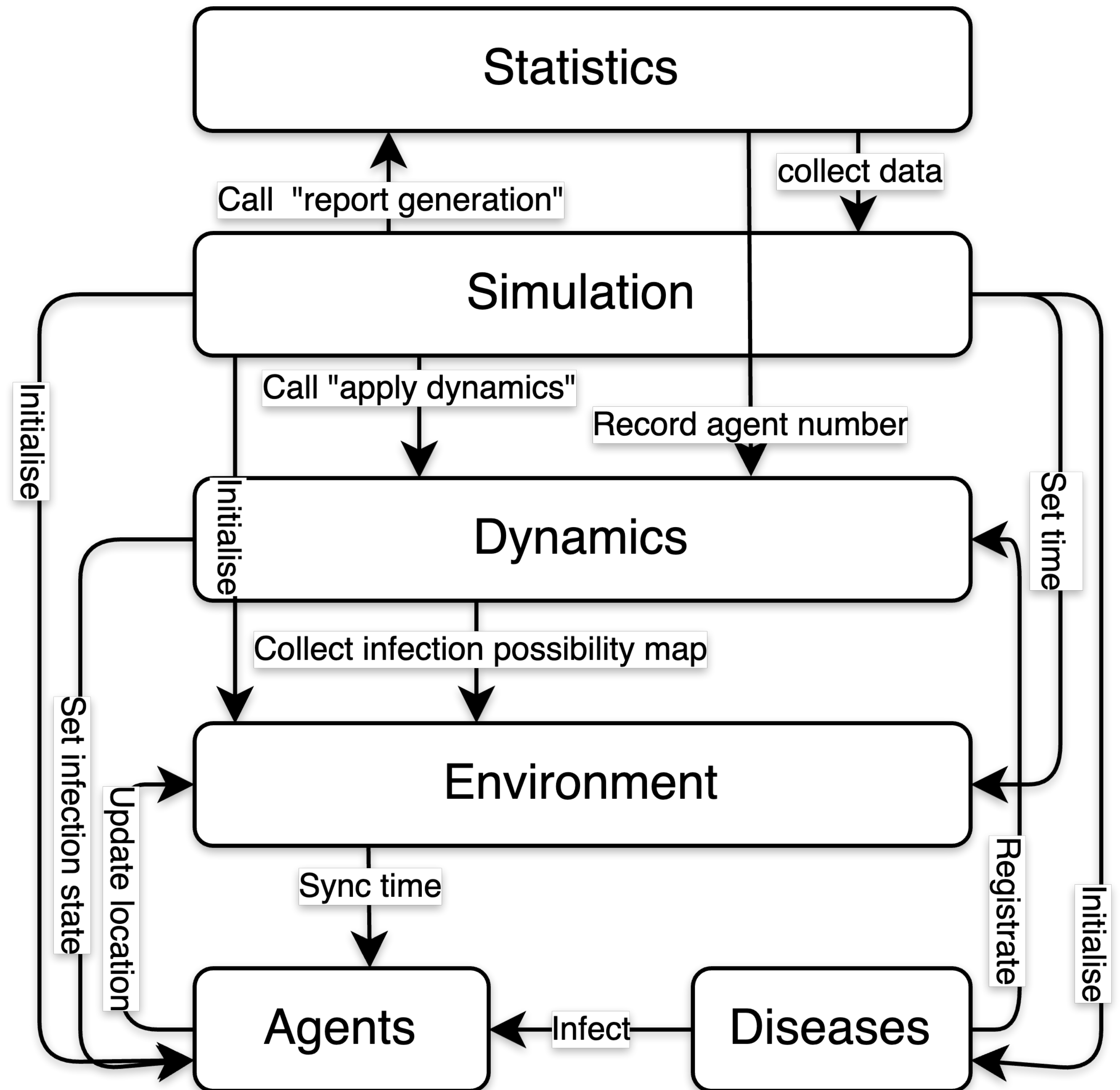


The plot for the possibility function

Model

How the modules work together

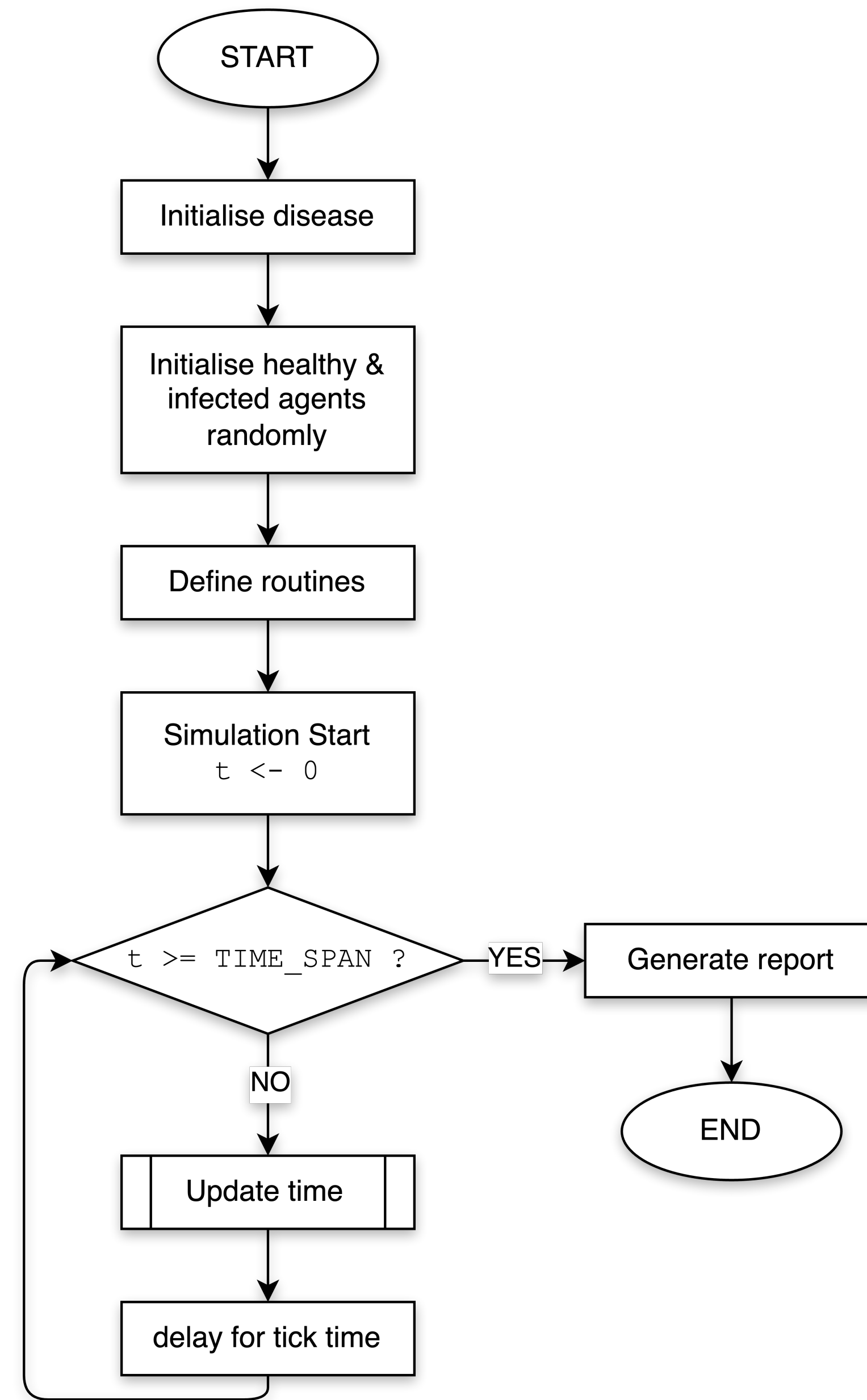
- This diagram shows how the **components** (modules) interact with each other.
- Each **rectangle** represents a **module**
- **Arrow** represents **action** done by the module to another.



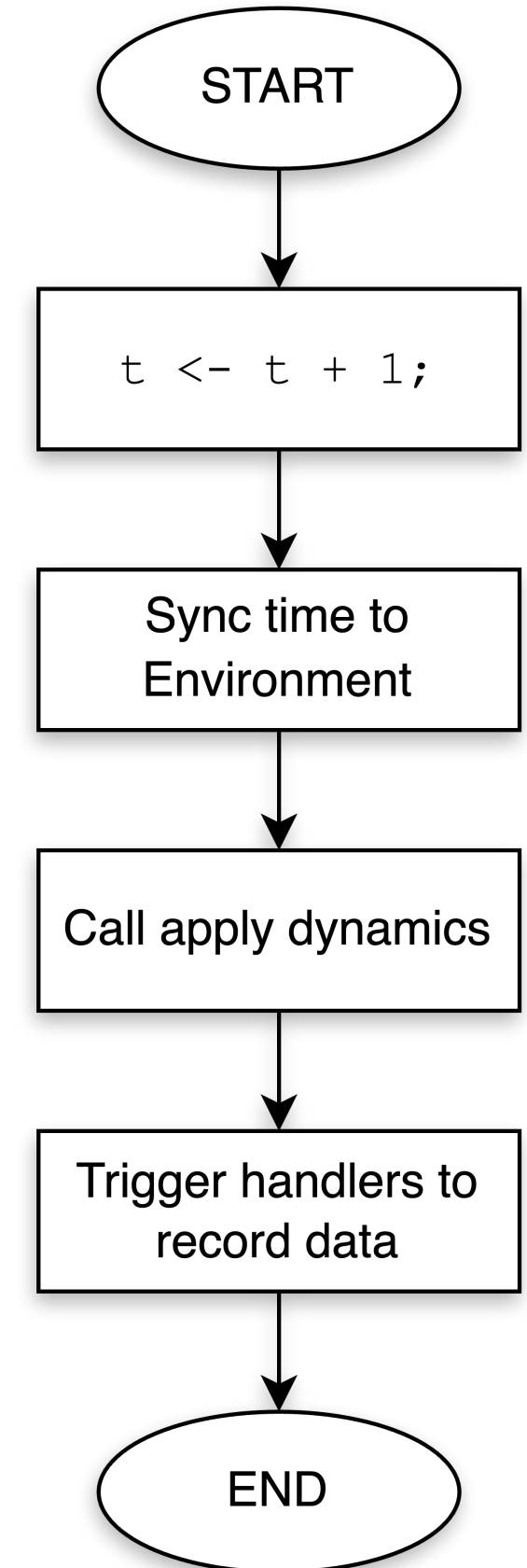
Model

How the model works

- The **flow chart** shows how the model works
- Major steps
 - **Initialisation**
 - **Simulation** by tick
 - **Generate report**



DEFINITION OF THE Update time STEP



The flow chart of the model

Implementation

Some coding works

- Programming languages used:
 - **TypeScript**: For **type safety** and **structure**.
 - More familiar with TypeScript
 - **R**: For **graphing** using ggplot2 library.



The Logo of TypeScript



The Logo of R

Results

The Outcomes from the Model with the Specific Parameters

Results

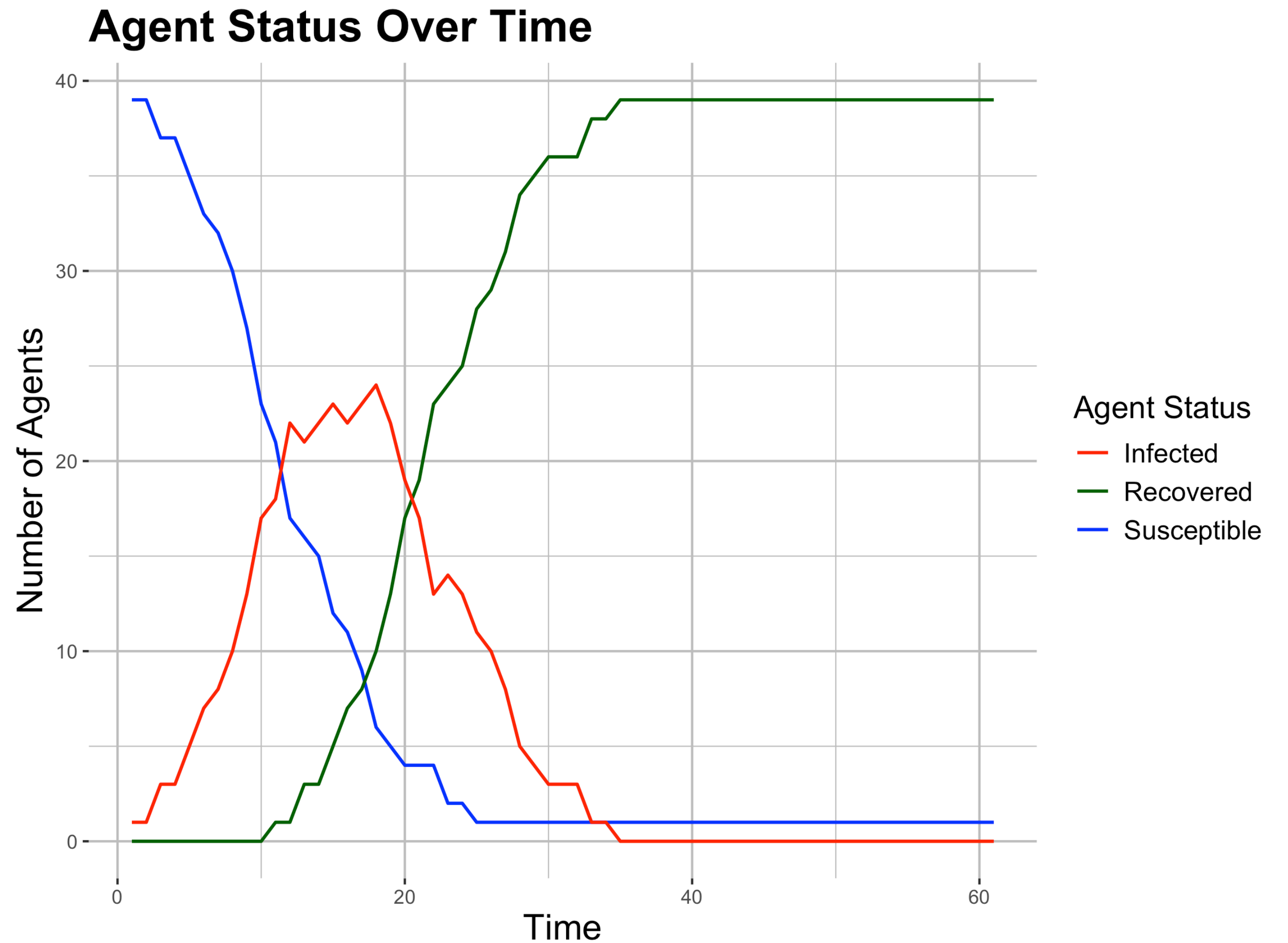
The parameters & diagram

- **Simulation Parameters:**

- Time: 60
- Agent Number: 40
- Room Size: 40

- **Disease Parameters:**

- Name: Cold
- Severity: 100
- Time to Restore: 10

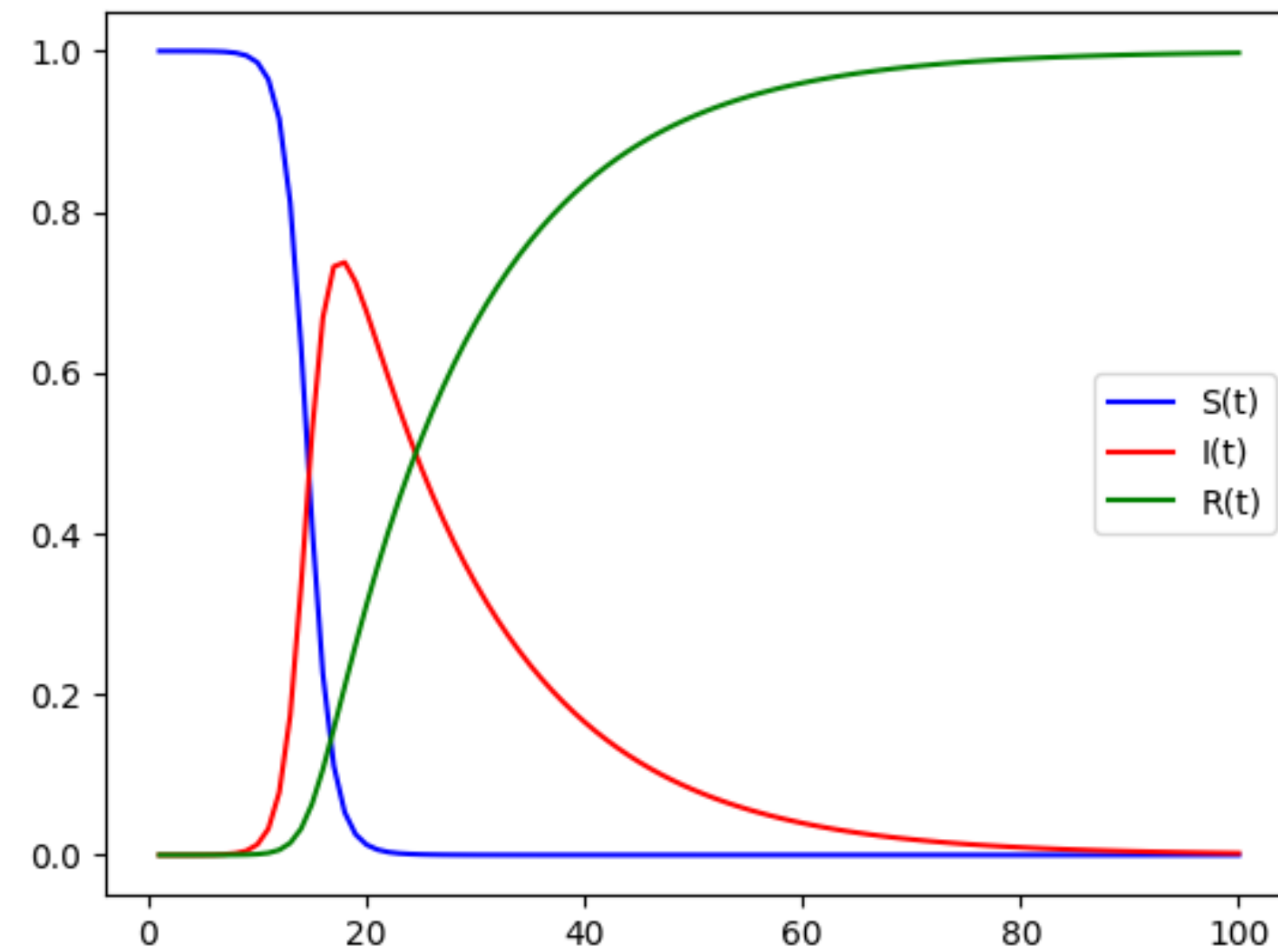


Results

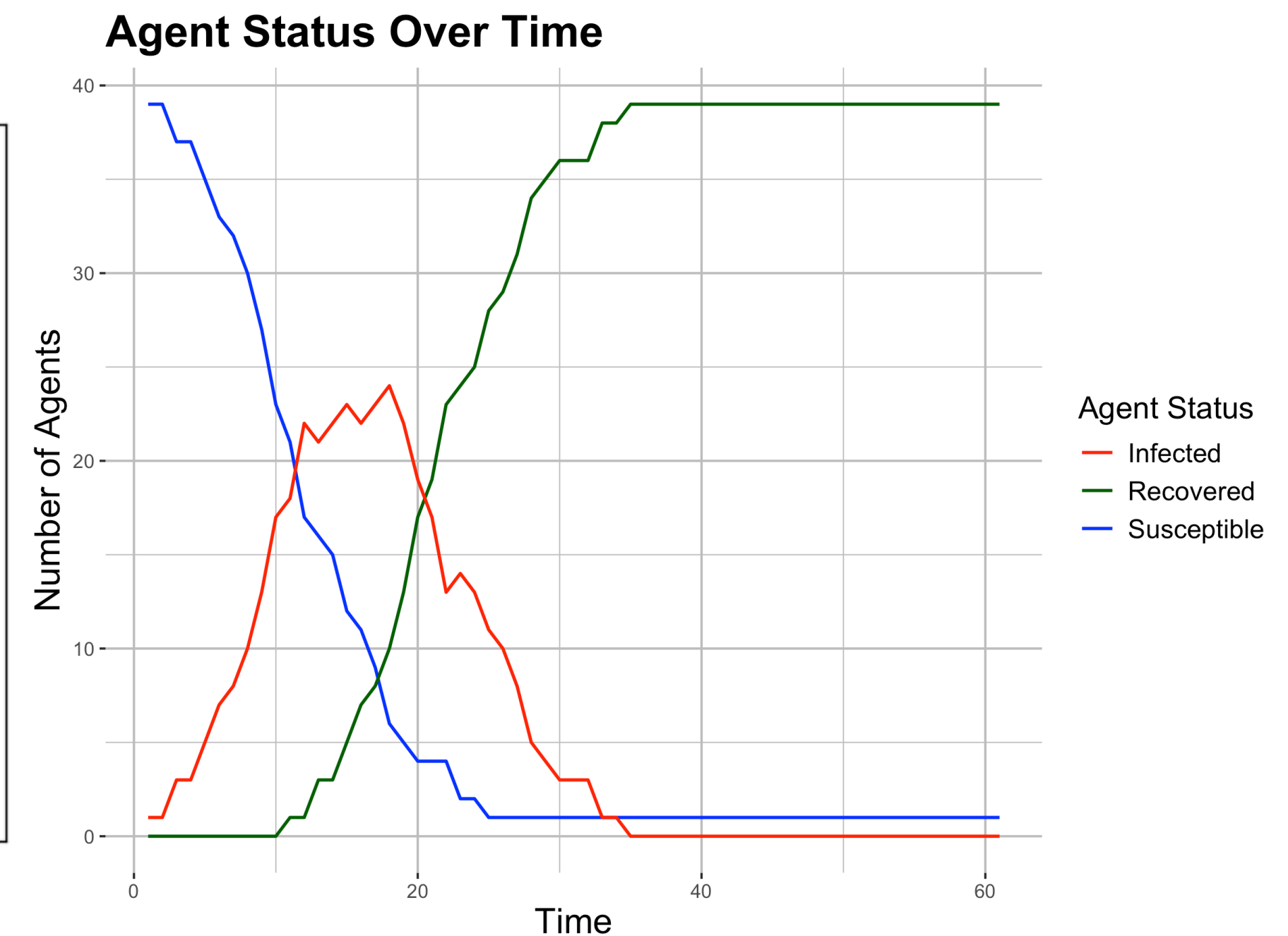
Compare the predictions from SIR model to this model

This model is:

- Capable of **partially** simulating the spread of the disease
- Due to the **similarity**
- Depict **outbreaks and extinctions**



The plot for SIR model, $\beta = 1$, $\gamma = 0.1$



The plot for previous agent based model

Discussion

Advantages & Drawbacks

Discussion

Advantages & Drawbacks

- **Strengths:**
 - Effective for **simple disease** spread simulation.
 - **Maintainable** code due to TypeScript.
- **Limitations:**
 - Lacks simulation of **multiple** diseases with **complex** dynamics.
 - Use recover **time**, not recover **possibility**
 - Lacks features of **migrating** parameters **from SIR model to this model**
 - Lacks simulation for some complex mechanisms between the infection possibility and the age, health of the agents, the severity of the disease, etc.

Conclusion

To Summarise the Insights from the Disease Prediction Model

Conclusion

To summarise

- Summary:
 - Model simulates disease **spread, outbreaks, and extinctions** accurately.
 - Future work includes **enhancing complexity and dynamic features**

Thanks for Listening

Thank You for Your Attention

The source code of the model is **available on Github**:

<https://github.com/Anson2251/camp-agent-base-model>

Heyan Zhu (Anson) Mathematics - Computer Science Class