Pandemia: Overview

James Thompson

October 2022

Introduction

Pandemia is an individual-based stochastic pandemic model, able to simulate the spread of an infectious disease over multiple regions, including the entire world.

The model is fast and scalable, able to simulate extremely large numbers of individuals.

This document presents an overview of the model.

The details of individual components are described in separate documents.

World

The **Pandemia** model acts on a **World**, with each **World** consisting of **Regions**, with each **Region** consisting of **Agents**, **Locations** and **Activities**.

Each agent performs a sequence of activities and performs these activities at particular locations.

For a world W containing a region R, let us denote by I, L and A the sets of agents, locations and activities associated to region R.

Then, denoting by T the time index set, to each agent $i \in I$ is associated a sequence of activities $\alpha_i : T \to A$ and a random variable $\lambda_i : A \times \Omega \to L$ determining the location $\lambda_i(a_i(t), \omega)$ of agent i at time t.

World

A **World** additionally consists of a **Travel Matrix**, representing how many agents travel from each region to each other region each day.

Here is the simplest possible **World** structure:

- One **Region**, one **Activity**, one **Location** and any number of **Agents**.

Here is the most complicated:

- Several **Regions**, with each **Region** consisting of several **Activities**, any number of **Locations** and any number of **Agents**, with a **Travel Matrix** describing the mixing between regions.

Components

The model features of a number of optional components:

- **Movement within Regions**
- **Movement between Regions**
- **Health**
- **Hospitalization**
- **Testing and Contact Tracing**
- **Vaccination**
- **Seasonality**

Additional or alternative components are easily added.

Pseudocode

```
for day in clock:
    movement_between_regions.dynamics(day)
    for region in regions:
        seasonality.dynamics(region, day)
        input.dynamics(region, day)
        for tick in range(ticks_in_day):
            t = (ticks_in_day * day) + tick
            health.dynamics(region, t)
            movement_within_regions.dynamics(region, t)
            hospitalization.dynamics(region, t)
        testing_and_contact_tracing.dynamics(region, day)
        vaccination.dynamics(region, day)
```

Interventions

The **Input** component allows the user to specify a **Policy**, consisting of interventions.

Possible interventions include the following:

- **Lockdown**
- **Border Closure**
- **Vaccination**
- **Testing and Contact Tracing**
- **Quarantine**
- **Face Masks**

Additional or alternative interventions are easily added.

Reporters collect output data for visualization and analysis.