GEORG-AUGUST-UNIVERSITÄT
GÖTTINGEN IN PUBLICA COMMODA
SEIT 1737

Project 6: Model for the Spread of Infectious Diseases

Carlo von Carnap

Summer Semester 2023

Final project *Computergestütztes wissenschaftliches Rechnen*,
Salvatore R. Manmana

Supervisor: Emily Klass

Inhaltsverzeichnis

1 Introduction **1**

2 Methodology **1**
2.1 Pseudorandom Number Generator (PRNG) `MT19937` 1

3 Implementation **1**
3.1 Usage of the Libraries `cvc_numerics.h` and `cvc_rng.h` 1
3.2 Structure and Workflow of the Main Program 1
3.3 Naming of Variables 2

4 Results and Discussion **2**
4.1 Model for the Spread of Infectious Diseases 2
4.2 Expected Ratio of Infected People averaged over Time 2
4.3 Vaccinated People without Participation in the Spread 2
4.4 Time Evolution of the Expected Ratio of Infected People 2

5 Supplementary Materials **6**

Bibliography **6**

1 Introduction

This model uses a quadratic grid of size $L \times L$ to examine the spread of an infectious agent. Each grid node is occupied by an immobile person and can be in one of the states Susceptible S, Infected I or Recovered R. Later, the forth state Vaccinated V is introduced, that permanently excludes the respective people from taking any of the three other states. For a person in one of the three active states S, I and R, the following three transitions are possible,

$$\text{Susceptible } S \xrightarrow{p_1} \text{Infected } I \xrightarrow{p_2} \text{Recovered } R \xrightarrow{p_3} \text{Susceptible } S.$$

With that, we have the three transition probabilities p_1, p_2 and p_3:

- p_1: a susceptible person getting infected by a direct infected neighbor
- p_2: an infected person turning recovered
- p_3: a recovered person returning susceptible

Vaccinated people V are set at the beginning of the simulation, with p_4 being the probability that a spot is occupied by one. The other three states are initializes with the same likelyhood. Each simulation step does now

2 Methodology

2.1 Pseudorandom Number Generator (PRNG) `MT19937`

For the generation of pseudorandom numbers the 1997 developed Mersenne Twister `MT19937` has been chosen.[1][2]

3 Implementation

3.1 Usage of the Libraries `cvc_numerics.h` and `cvc_rng.h`

3.2 Structure and Workflow of the Main Program

Generation of Random Numbers by a Static RNG

For the generation of pseudorandom numbers the in Subsection 2.1 described Mersenne Twister `MT19937` has been used. It was implemented statically and can therefore be accessed globally by the respective functions.

Implementation of the Modelling Grid

The above mentioned grid itself is realized as a $(L+2) \times (L+2)$ heap section of integer values, with L being the sidelength of the quadratic grid where the actual spread of the infection takes place. While this section is technically one-dimensional, it will for simplicity reasons be here refered to as a two-dimensional structure of the given shape. Inside the grid, the following integer values have been used to model the different states of the people within the simulation:

- `0`: this person is Susceptible S to the infection

- `1`: the person is Infected I
- `2`: the person is Recovered R and currently not susceptible
- `-1`: the person is Vaccinated V and does not participate in the spread

The grid has been implemented with an edge of ghosts at the top, bottom, left and right border, that are neither infectious nor subject to any updates of the grid — they will permanently take the value `0`.

Functions acting on the Modelling Grid

Structure of the Main Function and General Workflow

3.3 Naming of Variables

4 Results and Discussion

4.1 Model for the Spread of Infectious Diseases

4.2 Expected Ratio of Infected People averaged over Time

4.3 Vaccinated People without Participation in the Spread

4.4 Time Evolution of the Expected Ratio of Infected People

While previously the average of the ratio of infected individuals has been taken over time, the focus should now be layed on the time development of th einfection rate $\langle I \rangle_t$ for $N = 20$ samples. As grid size $L = 64$ was chosen for appropriate balance between running time and accuracy, the number of simulation steps again was set to $T = 1000$.

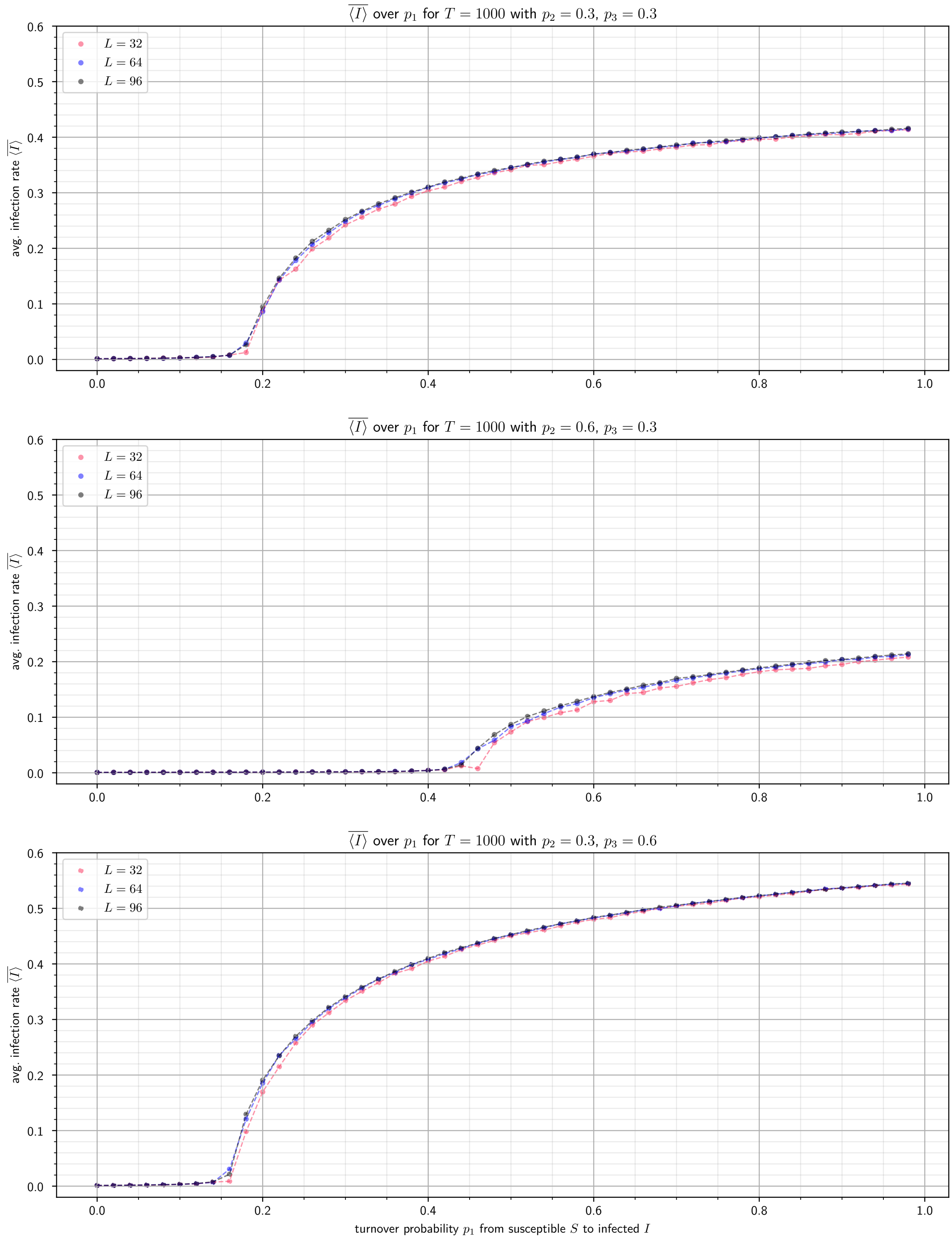

Abbildung 1: Graphic

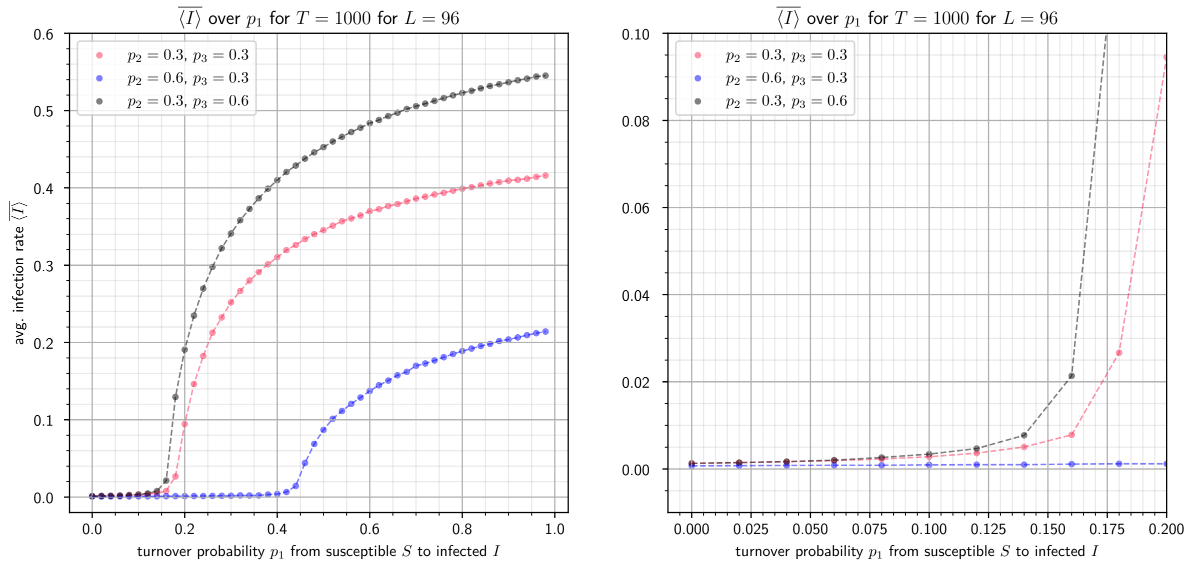

Abbildung 2: Graphic

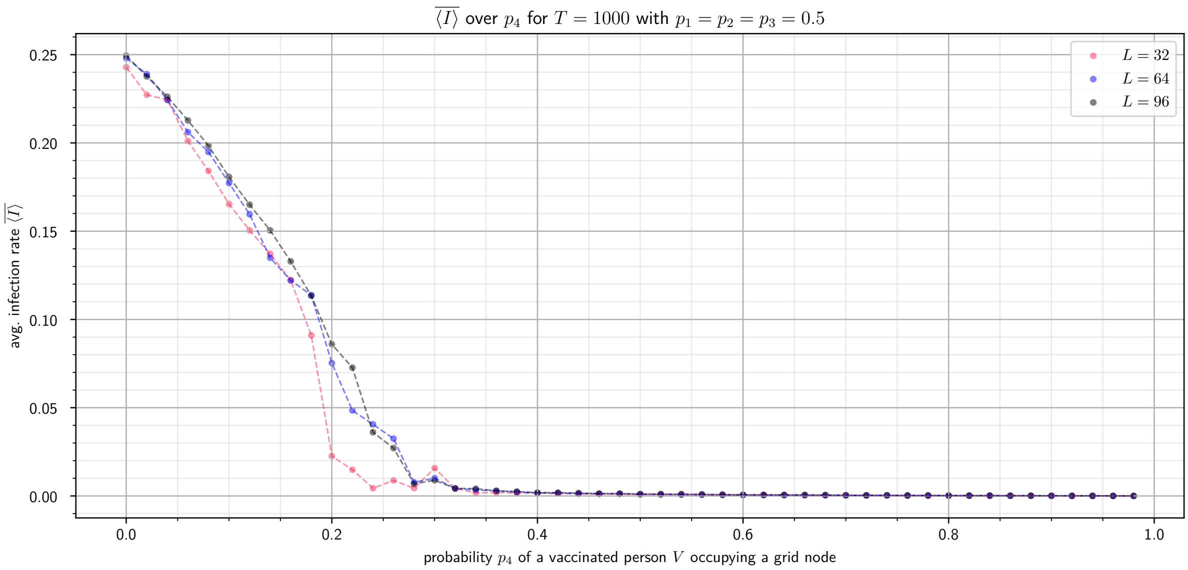

Abbildung 3: Graphic

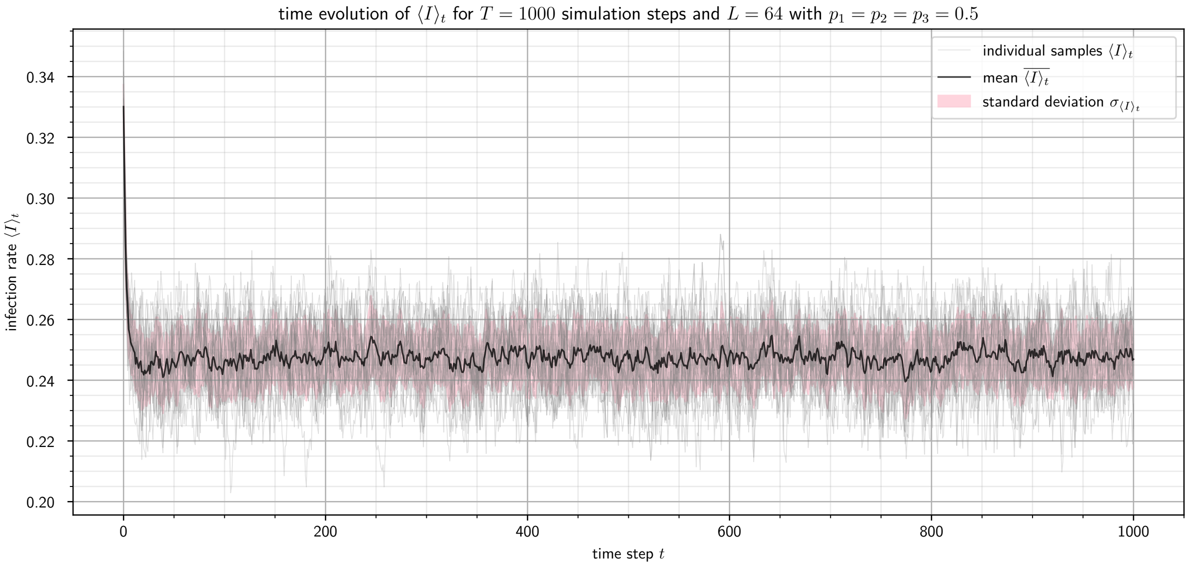

Abbildung 4: Graphic

5 Supplementary Materials

Bibliography

[1] *educative. What is Mersenne Twister?* 29. Juli 2023. URL: https://www.educative.io/answers/what-is-mersenne-twister (besucht am 29. 07. 2023).

[2] Makoto Matsumoto und Takuji Nishimura. „Mersenne Twister: A 623-Dimensionally Equidistributed Uniform Pseudo-Random Number Generator“. In: *ACM Trans. Model. Comput. Simul.* 8.1 (1998). ISSN: 1049-3301. DOI: 10.1145/272991.272995. URL: https://doi.org/10.1145/272991.272995.