

2. Homework (Complex Systems Block) Introduction to Focus Areas WS 2023/24

Deadline: December 2, 10:00 (**before** the lecture)

The homework should be worked out in groups. Pen & paper exercises will be discussed on the board. Programming exercises must be submitted via Whiteboard.

Homework 1 (Programming, 3+2 points)

You are given the following **epidemiologic model** of a virulent outbreak (also called susceptible-infected-recovered (SIR) model). Stoichiometric matrix S :

	r_1	r_2	r_3	r_4	r_5	r_6
X_1	1	-1	-1	0	0	0
X_2	0	0	1	-1	-1	0
X_3	0	0	0	0	1	-1
X_4	0	0	0	1	0	0

and propensities (reaction rates): $r_1 \dots r_6$.

$$r_1(t) = \lambda \tag{1}$$

$$r_2(t) = X_1(t) \cdot \delta \tag{2}$$

$$r_3(t) = X_1(t) \cdot X_2(t) \cdot \beta \tag{3}$$

$$r_4(t) = X_2(t) \cdot 3 \cdot 10^7 \cdot \delta \tag{4}$$

$$r_5(t) = X_2(t) \cdot k_r \tag{5}$$

$$r_6(t) = X_3(t) \cdot \delta \tag{6}$$

where, at each time instance t , X_1 denotes the number of susceptible individuals, X_2 are the number of infected individuals, X_3 are the number of individuals that recovered and are subsequently immune to infection and X_4 are the number of individuals that died from the infection. Parameter values are $\lambda = 1 \cdot 10^{-4}$, $\delta = 1 \cdot 10^{-8}$, $\beta = 5 \cdot 10^{-5}$, $k_r = 0.3$ and the initial state is $X_1(t_0) = \frac{\lambda}{\delta}$, $X_2(t_0) = 20$, $X_3(t_0) = 0$, $X_4(t_0) = 0$.

a) (**to be uploaded via Whiteboard**) Write a program implementing this model and generate trajectories using the stochastic simulation algorithm (SSA; also called Gillespie's algorithm). The program reads the input file ("Input.txt") provided in Whiteboard. The first number in the input file is the 'seed' of the random number generator, the second is the number of trajectories N to be computed (see explanations below). Using this input, compute the trajectories for N simulations up to time $T = 10$ (this means that the last time point you consider is $t \leq 10$). For each simulation, write the time and the values X_2 into a file "Task1TrajY.txt", where 'Y' = $1 \dots N$ is the current simulation (e.g. "Task1Traj3.txt" contains the jump-times and the trajectory of X_2 from the third simulation). The output textfile should be in the comma-separated text format using 3 digits after the comma (format '%1.3f'), e.g.

$$0.000, 0.010, 0.017, 0.202, \dots, 9.763 \tag{7}$$

$$5.000, 6.000, 5.000, 6.000, \dots, 12.000 \tag{8}$$

where the first row are the times $t^{(i)} \leftarrow t^{(i-1)} + \Delta t^{(i)}$ and the second row contains the corresponding number of *infected individuals* at that time $X_2(t^{(i)})$. Name your program "Exc2Task1.py"

and submit via Whiteboard.

Hint:

i) Remember the (extremely wise) hints given to you in the seminar.

ii) For Python users, to import and set the input-variables:

```
import numpy as np
```

```
In = np.loadtxt('Input.txt')
```

```
np.random.seed(seed=int(In[0]))
```

```
NrSimulations = int(In[1])
```

iii) To write the output:

```
states, times = SSA(...)
```

```
## save trajectory
```

```
Output = np.concatenate((np.array(times,ndmin=2),np.array(states,ndmin=2)), axis=0)
```

```
np.savetxt('traj'+str(i+1)+'.txt',Output,delimiter = ', ',fmt='%1.3f')
```

b) **(to be discussed in class)** Leave the initial state as it was, except that this time you set $X_2(t_0) = 2$. Perform $N = 1000$ simulations, generate a histogram of the number of X_2 at time $T = 10$ and depict the probability that 0, ..., 20 individuals are infected by $T = 10$.

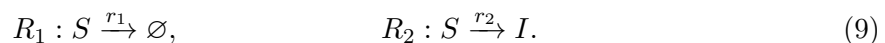
- What is the probability that the infection is still ongoing at $T = 10$?

From the $N = 1000$ simulations, generate a histogram of the number of X_4 at time $T = 10$ (number of casualties), depicting the probability that 0, ..., 35 individuals died by $T = 10$.

- How many individuals died on average (\pm standard deviation)?

Homework 2 (3 + 2 points)

Clinical trial simulation. Consider an infection-prevention-efficacy trial: A new intervention (vaccine) is tested, that intends to reduce the risk of infection with a deadly pathogen. The clinical trial can be described in terms of two reactions:



where S are the number of (susceptible) individuals that are (still) participating in the clinical trial and I are the number of individuals that got infected during the clinical trial. Consequently, reaction R_1 describes the 'drop-out', i.e. individuals that are no longer observed, because they did not show up any more for follow-up visits ('drop-outs'). Reaction R_2 describes an observed infection. I.e. an individual tested positive during the duration of the clinical trial. We have reaction propensities (*stochastic* reaction rates):

$$r_1 = k_{\text{dr-out}} \cdot S \quad (10)$$

$$r_2 = k_{\text{inf}} \cdot S \quad (11)$$

where k_{inf} denotes the (per-person) infection incidence and $k_{\text{dr-out}}$ denotes the (per-person) drop-out rate parameter.

Simulate the '**placebo arm**' of a COVID vaccine trial and plot the distribution of the number of infected individuals from $n_{\text{sim}} = 100$ stochastic simulations. Required parameters for your simulation can be deduced from the BionTech/Pfizer vaccine trial <https://doi.org/10.1056/NEJMoa2034577> (Table 2 therein, 'without evidence of infection').

1. Load 'Input.txt' and set the seed to the first number given therein.
2. Load the inversely sampled incidence parameters k_{inf} from the file 'kinf.txt' provided in whiteboard. For each instance of k_{inf} , calculate a corresponding 'drop-out' parameter using the following relation 'average follow-up time per person' = $\frac{1}{k_{\text{dr-out}} + k_{\text{inf}}}$.
3. Initialize your system with $S(t_0) =$ number of trial participants, $I(t_0) = 0$ and run a stochastic simulation, until $S(t) = 0$. Record the number of infection events. ($I(t_i)$, as well

as the times t_i when infection happen for task a) vs. the number of infected individuals $I(t_{\text{end}})$ at the end of each simulation for task b).

a) **(to be uploaded via Whiteboard)** Write a program implementing the clinical trial and simulate it using the stochastic simulation algorithm. Write the time of infection(s) t_i and the values of $I(t_i)$ into a file “Task2Infected.txt” using the same formatting as in Task 1. Call this program “Exc2Task2a.py” and upload via Whiteboard.

b) **(to be discussed in class)** Plot t_i vs. $I(t_i)$ as a step function. Then, perform $N = 100$ realisations of the clinical trial and plot a histogram of the number of infected individuals at the final time $I(t_{\text{end}})$ that you computed from each respective simulation. Mark the sample average of infected individuals in the same figure,. Also mark the number of cases reported in the clinical trial paper <https://doi.org/10.1056/NEJMoa2034577> (Table 2).

- **Discuss:** Why did you get a (slightly) different number than in the clinical trial?
- **Discuss:** What does that tell you and which information have you gained that you could use to evaluate the clinical trial?

good luck! ...