# Numerical Simulation of SIS dynamics on Complex Networks

Peter G. Fennell

## Contents

# 1 About

Numerical Simulation of Susceptible-Infected-Susceptible (SIS) dynamics on complex networks. Possible simulation algorithms are the Gillespie Algorithm (event-based algorithm), Asynchronous updating and Synchronous updating (discrete-time algorithms). The time-step for the discrete-time algorithms is automatically set in the source code (S.h and AS.h) and can be modified if required.

If used in publications, please cite:

[1] Fennell, P.G., Melnik, S, Gleeson, J.P., "The limitations of discrete-time approaches to continuous-time contagion dynamics" (2016).

# 2 Compiling

To compile the executable "sim" used to perform the simulations:

1. Open a terminal/command prompt window
2. Set the working directory to this folder
3. Perform the command "make sim"

Note: alternative to step 3, the code can be manually compiled to include optimisation flags such as -02 as "g++ -o sim -02 main.cpp"

# 3 Performing Simulations

The following is a description of how to perform the simulations on different network types including 1) Erdős-Rényi graphs, 2) Complete graphs and 3) custom graphs specified by an edge list.

## 3.1 Erdős-Rényi (ER) Graph

To perform simulations on an ER Graph run the following command from the terminal in the same location as the executable file "sim" is located:

```
./sim sim beta mu X0 1 N z tend N_trials Xdat EXdat PXTdat
```

### 3.1.1 Explanation of variables

- sim: The simulation method to use. Takes values 1 for GA, 2 for AS and 3 for S.

- beta: The SIS transmission rate. It can take any non-negative value
- mu: The SIS recovery rate. It can take any non-negative value
- X0: The number of nodes in the network that are initially infected
- 1: Indicates to perform simulations on an ER network
- N: The number of nodes in the networks
- z: The mean degree of the ER network
- tend: The end time i.e. the length of time that the SIS dynamics should be simulated for
- N_trials: The number of stochastic trials to perform
- Xdat: Takes values either 1 or 0; 1 indicates to the program to save data from each of the N_trials trials, 0 indicates not to save*.
- EXdat: Takes values either 1 or 0; 1 indicates to the program to save the data average values over each of the N_trials trials, 0 indicates not to save*.
- PXTdat: Takes values either 1 or 0; 1 indicates to the program to save the probability distributions formed from data from each of the N_trials trials, 0 indicates not to save*

*For further inputs on the program outputs, see section ”Outputs” below

#### 3.1.2 Example

The following commands produces the data of the type used in Figure 4 of [1] for the Gillespie Algorithm and Synchronous updating simulations. That case was SIS dynamics on an ER network of 1000 nodes and mean degree z=4 with beta = 0.2 and mu = 0.72.

```
./sim 1 0.2 0.72 10 1 1000 4 200 100 0 1 0
./sim 3 0.2 0.72 10 1 1000 4 200 100 0 1 0
```

The expected value curves are in the output files ”data/GA_EX_ER_N1000_m0.72_b0.2.txt” and ”data/S_EX_ER_N1000_m0.72_b0.2.txt” respectively.

## 3.2 Complete Graph

To perform simulations on a Complete Graph (one in which every node is connected to every other node) run the following command from the terminal in the same location as the executable file "sim" is located:

```
./sim sim beta mu X0 2 N tend N_trials Xdat EXdat PXTdat
```

### 3.2.1 Explanation of variables

- sim: The simulation method to use. Takes values 1 for GA, 2 for AS and 3 for S.
- beta: The SIS transmission rate. It can take any non-negative values
- mu: The SIS recovery rate. It can take any non-negative values
- X0: The number of nodes in the network that are initially infected
- 2: Indicates to perform simulations on a complete graph
- N: The number of nodes in the networks
- tend: The end time i.e. how the length of time that the SIS dynamics should be simulated for
- N_trials: The number of stochastic trials to perform
- Xdat: Takes values either 1 or 0; 1 indicates to the program to save data from each of the N_trials trials, 0 indicates not to save*.
- EXdat: Takes values either 1 or 0; 1 indicates to the program to save the date average values over each of the N_trials trials, 0 indicates not to save*.
- PXTdat: Takes values either 1 or 0; 1 indicates to the program to save the probability distributions formed from data from each of the N_trials trials, 0 indicates not to save*

*For further inputs on the program outputs, see section "Outputs" below

### 3.2.2 Example

The following commands produce the data used in Figure 3 of [1] for the Gillespie algorithm and Synchronous updating respectively. That case was SIS dynamics on a fully connected network of 10 nodes with beta and mu both equal to 1 and one node initially infected

```
./sim 1 1 1 1 2 10 3 100000 0 1 1
./sim 3 1 1 1 2 10 3 100000 0 1 1
```

The expected value curves that appear in Figure 3a are in the output files ”data/GA_EXdat_CG_N10_m1_b1.txt” and ”data/AS_EXdat_CG_N10_m1_b1.txt” respectively.

The probability distributions that appear in Figure 3b and Figure 3c are in the output files ”data/GA_PXTdat_CG_N10_m1_b1.txt” and ”data/S_PXTdat_CG_N10_m1_b1.txt” respectively.

## 3.3 Custom Network

For a custom network, the input to the programme is a text file “filename” consisting of an edge list. To perform simulations on the custom network, run the following command from the terminal in the same location as the executable file “sim” is located:

```
./sim beta mu X0 3 filename tend N_trials Xdat EXdat PXTdat
```

### 3.3.1 Explanation of variables

- beta: The SIS transmission rate. It can take any non-negative values
- mu: The SIS recovery rate. It can take any non-negative values
- X0: The number of nodes in the network that are initially infected
- 3: Indicates to perform simulations on a complete graph
- filename: The name of the file in which the edge list is stored. This should be the files address from the current directory.
- tend: The end time i.e. how the length of time that the SIS dynamics should be simulated for
- N_trials: The number of stochastic trials to perform
- Xdat: Takes values either 1 or 0; 1 indicates to the program to save data from each of the N_trials trials, 0 indicates not to save*.

- EXdat: Takes values either 1 or 0; 1 indicates to the program to save the data average values over each of the N_trials trials, 0 indicates not to save*.
- PXTdat: Takes values either 1 or 0; 1 indicates to the program to save the probability distributions formed from data from each of the N_trials trials, 0 indicates not to save*

*For further inputs on the program outputs, see section "Simulation Outputs" below

### 3.3.2 Example

The following commands produce Gillespie algorithm, Asynchronous updating and Synchronous updating simulations for SIS dynamics on a custom network with beta and mu both equal to 1 and one node initially infected. The network here is a simple chain network of 5 nodes, and the edge list is stored in the file chain.txt

```
./sim 1 1 1 1 3 chain.txt 200 100 0 1 0
./sim 2 1 1 1 3 chain.txt 200 100 0 1 0
./sim 3 1 1 1 3 chain.txt 200 100 0 1 0
```

# 4 Simulation Outputs

The "Xdat", "EXdat" and "PXTdat" 1/0 flags dictate what output the program saves. Output files are saved in the directory 'data/'

## 4.1 Xdat

In each simulation, the fraction of infected nodes over time is recorded. Recordings are taken at 101** points throughout the simulation, beginning at time t=0 and ending at time t=tend. If the flag Xdat is set equal to 0, then these 101 recordings - for each of the N_trials realizations - are stored in an output file "_Xdat_.txt". In the output file, each row is for a time point while each column is for a different realisation i.e.

```
x(t=0, trial=1) x(t=0, trial=2) ...x(t=0, trial=N_trials)
x(t=t1, trial=1) x(t=t1, trial=2) ...x(t=t1, trial=N_trials)
...
x(t=tend, trial=1) x(t=tend, trial=2) ...x(t=tend, trial=N_trials)
```

** If more/less than 101 points are desired then the number can be set by changing the variable n_intervals in the source code file "main.cpp" and recompiling as described in the section "EDITING SOURCE CODE" below.

## 4.2 EXdat

The expected fraction of infected nodes over time can be computed by averaging the data described in ”Xdat” over all of the realizations. If the flag EXdat is set equal to 1, then this expected fraction is saved in an output file ”_EXdat”, again beginning at time t=0 and ending at time t=tend. In the output file, there is one column and the time evolution is along the rows i.e.

Ex(t=0)
Ex(t=t1)
. . .
Ex(t=tend)

## 4.3 PXTdat

This file stores the probability that there are X infected nodes at time t. This is done by counting the number of trials in which there are X infected individuals at time t, and computing the empirical probability density functions. If the flag PXTdat is equal to 1, then this empirical probability density function is stored in the output file ”_PXTdat” in the form

p(X=0,t=0) p(X=1,t=0) . . . p(X=N,t=0)
p(X=0,t=t1) p(X=1,t=0) . . . p(X=N,t=1)
. . .
p(X=0,t=tend) p(X=1,t=tend) . . . p(X=N,t=tend)

# 5 Editing source code

The source code files ”main.cpp”, ”main.h”, ”GA.h”, ”AS.h” and ”S.h can be edited to suit the preferences of users. After the source code files have been edited, the executable ”sim” can be compiled as described in Section 2: Compiling