Homework 2

Francesca Cossu, s305746

December 11, 2022

Exercise 1

The first part of this assignment consist in studying a single particle performing a continuous-time random walk in the network described by the graph in figure 1 and with the following transition rate matrix:

$$\Lambda = \begin{pmatrix} 0 & 2/5 & 1/5 & 0 & 0 \\ 0 & 0 & 3/4 & 1/4 & 0 \\ 1/2 & 0 & 0 & 1/2 & 0 \\ 0 & 0 & 1/3 & 0 & 2/3 \\ 0 & 1/3 & 0 & 1/3 & 0 \end{pmatrix}$$

The task consist in simulating the particle moving around in the network in continuous time according to the transition rate matrix. In the CTMC the time is a continuum, but the random process still describes the evolution of a state variable x inside a discrete space C with a graph structure. The transition rate matrix Λ describes all possible transitions between nodes/states.

Transition happens at random time instants decided by the tick of the Poisson Clock; the latter is characterized by the property that the time elapsed between any two of its consecutive ticks is an independent random variable with exponential distribution with a specified rate.
In order to simulate a Poisson clock with rate r, one must simulate the time between two consecutive Poisson clocks, denoted by $t_{next} = -ln(u)/r$ where u is a random variable with uniform distirbution, $u \in \upsilon(0, 1)$.

To model the CTMC I equipped each node with its own Poisson clock with rate $\omega_i = \sum_j \Lambda_{ij}$: when the particle is at node i and the clock of that node ticks, the particle jump to a neighbor j with probability $P_{ij} = \frac{\Lambda_{ij}}{\omega_i}$.

Question a

What is, according to the simulations, the average time it takes a particle that starts in node a to leave the node and then return to it?

I computed the average time it takes the particle a to leave the node and return to it by creating a vector with the sum of all transition times for each complete walk from a to a and then sum all the return times and dividing by the length of the vector. The average return time found is $5.69s$.

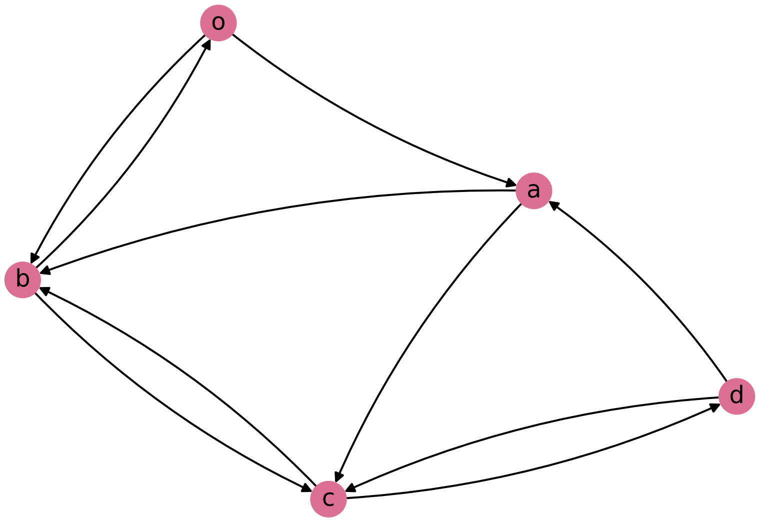

Figure 1:

Question b

How does the result in a) compare to the theoretical return-time $\mathbb{E}_a[T_a^+]$? (Include a description of how this is computed.)

From the theory is known that the expected return time is computed with the formula:

$$\mathbb{E}_a[T_a^+] = \frac{1}{\omega_i \bar{\pi}_i}$$

I calculated the theoretical return time with the formula and this computing gave the result 6.75.

Question c

What is, according to the simulations, the average time it takes to move from node o to node d?

I calculated the hitting time with the same method I used for point a. The average hitting time found is 8.33.

Question d

How does the result in c) compare to the theoretical hitting-time $\mathbb{E}_0[T_d]$? (Describe also how this is computed.)

The expected hitting time in continuous Markov chain related to destination set S is:

$$\mathbb{E}_i[T_S] = \frac{1}{\omega_i} + \sum_j P_{ij}\mathbb{E}_j[T_S] \qquad \textit{for } i \notin S$$

$$\mathbb{E}_i[T_S] = 0 \qquad \textit{for } i \in S$$

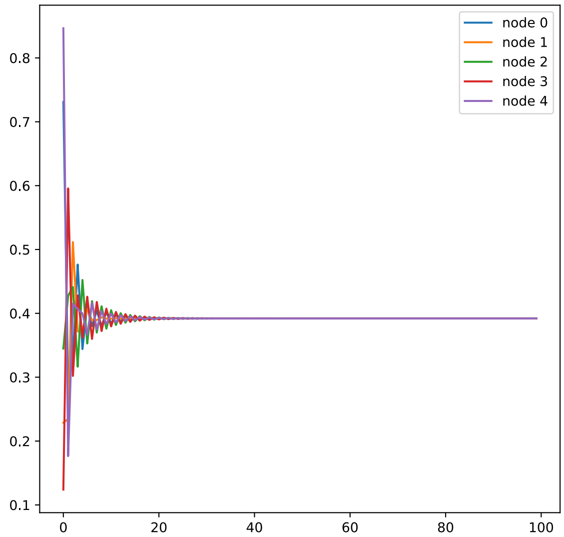

Figure 2:

Considering $\tau_i = \mathbb{E}_i[T_S]$, τ can be computed solving the system $\tau = (I - \hat{P})^{-1}\hat{\omega}$ where $\hat{P}$ is obtained by removing the rows and columns corresponding to the nodes in the set S from P, while $\hat{\omega} = \frac{1}{\omega_i}$ for all i $\notin$ S.

As expected the computed time is similar: $\mathbb{E}_0[T_d] = 8.80$

Question e

Interpret the matrix Λ as the weight matrix of a graph G = (V,E,Λ), and simulate the French-DeGroot dynamics on G with an arbitrary initial condition x(0). Does the dynamics converge to a consensus state for every initial condition x(0)? Motivate your answer.

This is a problem in the context of social learning. We assume that a community of agents is connected by a graph G = (V,E,Λ) that is strongly connected and aperiodic. A social aggregation of information takes place under a linear averaging dynamics $x(t + 1) = Px(t)$, known as French-DeGroot dynamics, where P is the normalized adjacency matrix and $x(t)$ is the vector of opinions during time.
To simulate the French-DeGroot dynamics on the graph in figure 1, random initial conditions $x(0)$ have been taken into account, and in all the cases analyzed, consensus is reached. In figure 2 there is an example of an evolution of consensus.

Question f

Assume that the initial state of the dynamics for each node $i \in v$ is given by $x_i(0) = \xi_i$, where $\{\xi_i\}_{i\in v}$ are i.i.d random variables with variance σ^2. Compute the variance of the consensus value, and compare your results with numerical simulations.

In this exercise the system's initial condition is not fixed. Every component in the state vector is a random variable with variance equal to ξ^2. For this exercise there is a random variable $x_i(0)$ with uniform distribution:

$$x_i(0) = \xi_i \qquad \xi_i \backsim U(0,1)$$

Since we use $U(0,1), Var[\xi_i] = \frac{1}{12}$.
The variance of the final consensus is computed using:

$$Var[\alpha] = \mathbb{E}[(\alpha - \mathbb{E}[\alpha])^2]$$

and, since $\alpha = \pi' x(0)$:

$$Var[\alpha] = Var[x_i(0)] \sum_i \pi_i^2 = \sum_i \frac{\pi_i^2}{12}$$

Concluding, I found that $Var[\alpha] = 0.0178$

Question g

Remove the edges (d,a) and (d,c). Describe and motivate the asymptotic behaviour of the dynamics. If the dynamics converges to a consensus state, how is the consensus value related to the initial condition x(0)? Assume that the initial state of the dynamics for each node i $\in v$ is given by $x_i(0) = \xi_i$, where $\{\xi_i\}_{i \in v}$ are i.i.d random variables with variance σ^2. Compute the variance of the consensus value. Motivate your answer.

In this case the consenus value is the average of the initial conditions of the nodes, where the weights are given by the invariant distribution π. In the graph only the node d belongs to the sink of the condensation graph, so the invariant distribution centrality is 0 for all the other nodes. Since $\pi' x(0) = \alpha$, the consensus converges to the initial opinion of agent d. Concluding, I found that $Var[\alpha] = 0.083$

Question h

Consider the graph $(v, \varepsilon, \Lambda)$, and remove the edges (c, b) and (d, a). Analyse the French-DeGroot dynamics on the new graph. In particular, describe and motivate the asymptotic behaviour of the dynamics in terms of the initial condition x(0).

Removing the edges (c,b) and (d,a), the resulting graph has a new sink in the condensation graph containing nodes c and d, as it is visible in figure 3. Node c and d are the *trapping component*. The graph obtained is periodic, so the consensus is reached only if c and d have the same initial opinion. An example of an evolution is visualable in figure 4

Exercise 2

In this part we will again consider the network of exercise 1, with weights according to (1). However, now we will simulate many particles moving around in the network in continuous time. Each of the particles in the network will move around just as the single particle moved around in Problem 1: the time it will stay in a node is exponentially distributed, and on average it will stay $\frac{1}{\omega_i}$ time-units in a node i before moving to one of its out-neighbors. The next node it will visit

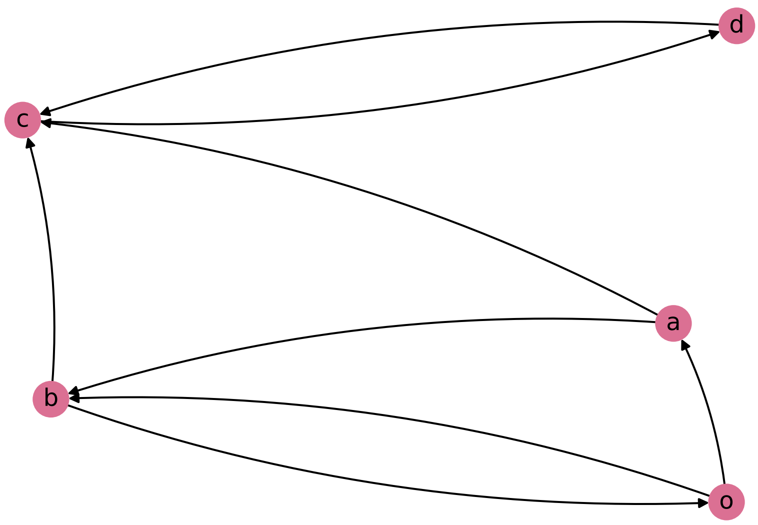

Figure 3: Network ex 1.g

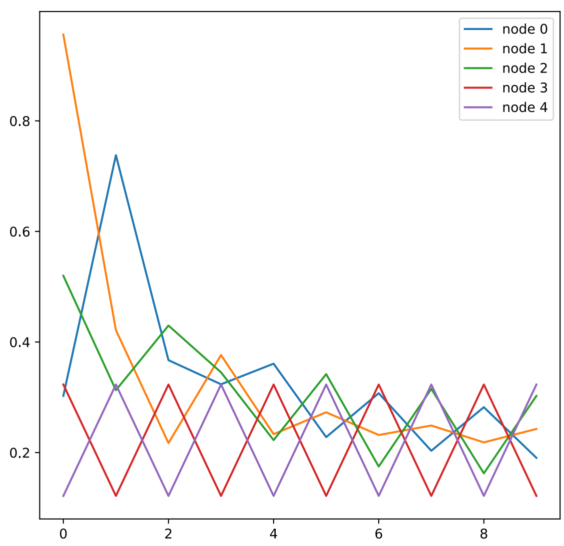

Figure 4: Evolution of consensus dynamic (ex 1.g)

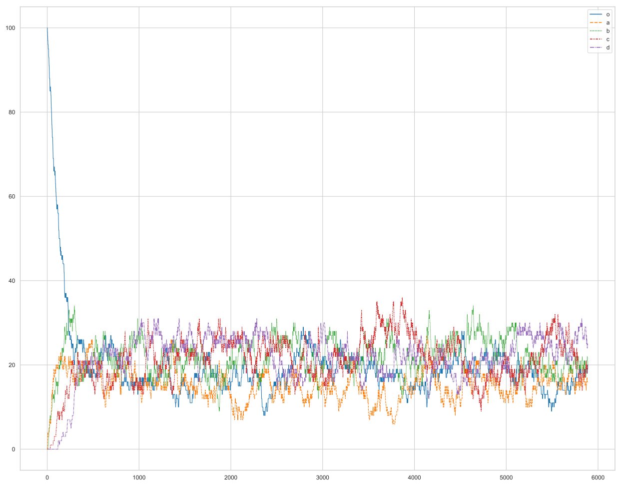

Figure 5: Evolution of number of particle in the nodes

is based on the probability matrix $P = diag(\omega)_{-1}\Lambda$, where $\omega = \Lambda$. Your task is to simulate this system from two different perspectives: the particle perspective, i.e. "follow the particle", and the node perspective, i.e. "observe from the node". The departure times of the node can thus be seen as a Poisson process with rate $N_i(t)\omega_i$. At each tick of the Poisson clock of the node, it will move a particle to a neighboring node. The node to which the particle will move is again based on the normalized transition rate matrix P.
Particle perspective: If 100 particles all start in node a, what is the average time for a particle to return to node a?
How does this compare to the answer in Problem 1, why?

This part of the exercise was handled with the same scheme of exercise 1.a because the movement of a particle has no impact on the others' behaviour. The average return time found was the same as exercise 1.a.

Node perspective: If 100 particles start in node o, and the system is simulated for 60 time units, what is the average number of particles in the different nodes at the end of the simulation?
in this step of the exercise the focus was the amount of particle within each node over time. The rate of each node in this case is:

$$t_{next} = -\frac{ln(u)}{N_i(t)\omega_i}$$

For this task, differently for the others, is used a global clock with rate $N_{net}\omega^{\cdot}$ where N_{net} is the total number of particles in the network and $\omega^{\cdot} = max(\omega_i)$. For each step of the simulation a node from which move a particle at each tick was chosen at random and in proportion to the number of particle in every node. Then a particle from the chosen node will jump in accordance with the matrix of the transition probability Q. The evolution of the number of the particles in each node is shown in figure 5 The average particle distribution is:

$$N_i = [18.68,\ 14.42,\ 22.66,\ 21.86,\ 21.38]$$

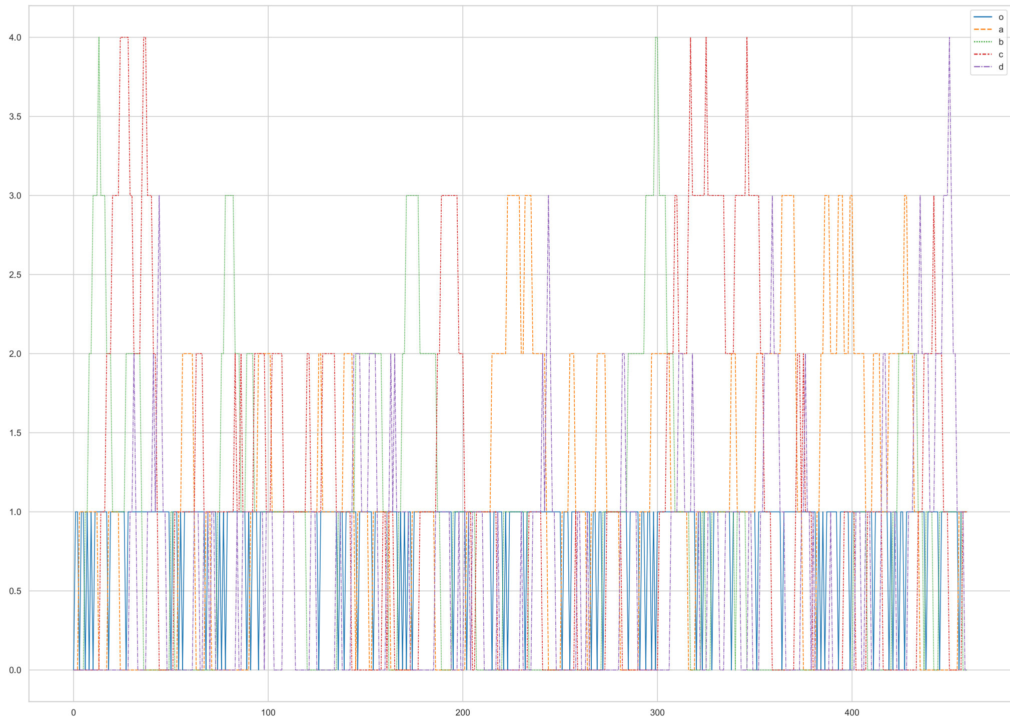

Figure 6: Evolution of number of particle in the nodes, proportional rate

Exercise 3

In this part we study how different particles affect each other when moving around in a network in continuous time. We consider the open network with transition rate matrix Λ_{open}:

$$\Lambda_{open} = \begin{pmatrix} 0 & 3/4 & 3/8 & 0 & 0 \\ 0 & 0 & 1/4 & 1/4 & 2/4 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

For this system, particles will enter node o according to a Poisson process with input rate $\lambda = 1$.

Proportional rate: each node i will pass along particles according to a Poisson process with rate equal to the number of particles in the node times the rate of the local Poisson clock of node.

In this part of the exercise, as the exercise 2 from the node perspective, is assumed that each node will pass along particles at a rate proportional to the number of particles contained in it. The distribution of particles across nodes for a simulation of 60 time units is presented in figure 6

What is the largest input rate that the system can handle?

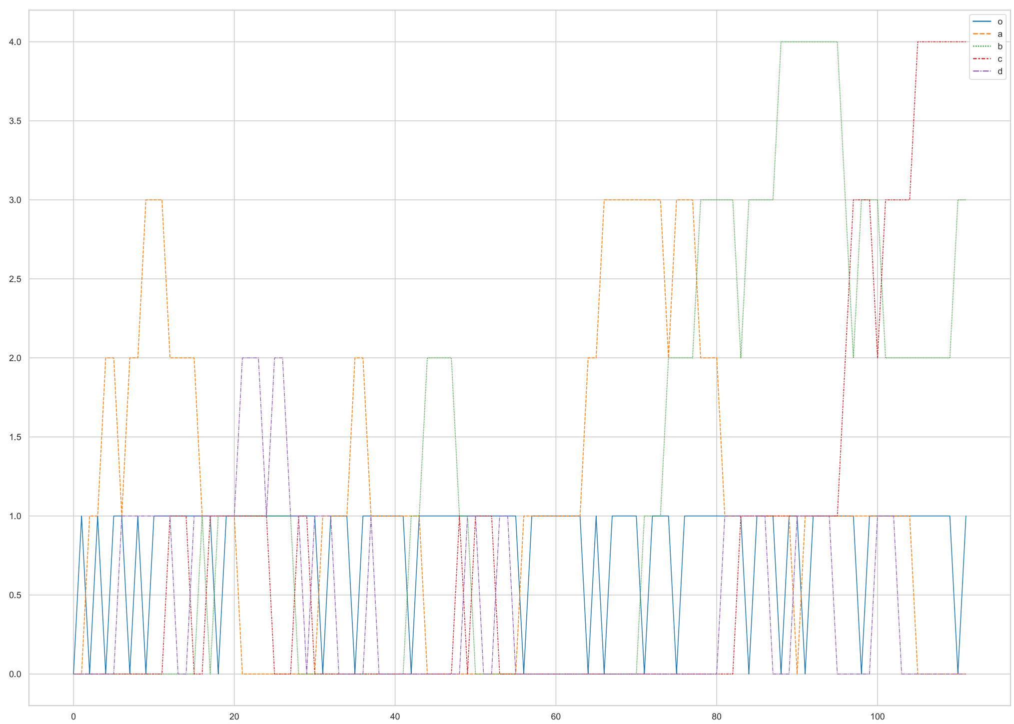

Figure 7: Evolution of number of particle in the nodes, fixed rate

The rate of a node is proportional to the number of particles in that node, so with a large input rate there will be a large number of particles in the network and they will also move inside the graph, so the system can handle a large input rate with a proportional rate. *Fixed rate: Simulate the system for 60 time units and plot the evolution of number of particles in each node over time.*

The distribution of particles across nodes for a simulation of 60 time units is presented in figure 7.

What is the largest input rate that the system can handle without blowing up? Why is this different from the other case? For an input rate lesser than ω_o the network can handle the particles.