Report Homework 2

Chiara Scagliola
Politecnico di Torino
s303037@studenti.polito.it

Chiara Van der Putten
Politecnico di Torino
s306273@studenti.polito.it

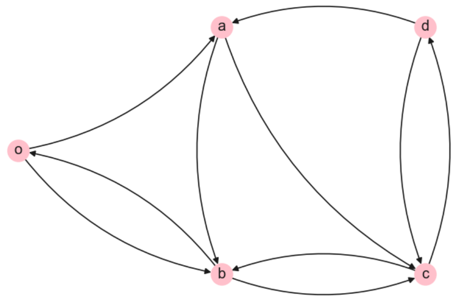
December 2022

1 Exercise

1.1 Simulation of a particle in a network in continuous time

In this first part, it's simulated the particle moving around in the network (Figure 1) in continuous time.

Figure 1: Graph representing the network



To compute the average time it takes a particle that starts in node a to leave the node and then return to it, a unique global Poisson clock it's defined, with rate $\omega^* = \max_i(\omega_i)$ where $\omega_i = \sum_j \Lambda_{ij}$ and Λ represents the transition rate matrix. After each Poisson clock ticks the particle jumps to a neighbor j with probability $Q_{ij} = \frac{\Lambda_{ij}}{\omega_*}$, $i \neq j$ or remain in the same node with probability $Q_{ii} = 1 - \sum_{i \neq j} Q_{ij}$. To simulate the Poisson process, the time t_{next} between two ticks of the Poisson clock is calculate in the follow way: $t_{next} = \frac{-ln(u)}{\omega^*}$ Where u is a uniformly distributed random variable U(0, 1). To obtain the average return time the experiment was simulated for 100.000 times obtaining the following result : 6.755. The theoretical return-time is equal to:

$$\mathbb{E}_a[T_a^+] = \frac{1}{\pi_a} = \frac{1}{\bar{\pi}_a \omega_a} = 6.750$$

where $\bar{\pi}_a(t) = \mathbb{P}(X(t) = a), \quad i \in \mathcal{X} = \{o, a, b, c, d\}$. The average time that the particle takes to move from node o to node d has been calculated with the same algorithm of the previous exercise, the result obtained is equal to 8.810. The theoretical hitting-time is calculated with the following formula:

$$\mathbb{E}_o[T_S] = \frac{1}{\omega_o} + \sum_j P_{oj} \mathbb{E}_j[T_S]$$

where $S \subseteq V$ and $V = \{d\}$ $\mathbb{E}_o[T_S]$ for all the nodes $R = V \backslash S$ can be calculated by solving the system:

$$\tau = \frac{1}{\omega} + P_{|RxR}\tau$$

where $P_{|RxR}$ is obtained by removing from P the rows and the columns associated with S. In our case S contains only the destination node d, τ can be rewritten as:

$$\tau = (I - Q)^{-1}\frac{1}{\omega}$$

the result represents the hitting time of each node to reach d: [8.78 7.142 7.071 3.357] as can be seen the hitting time from o to d is equal to 8.78, which is similar to the result obtained previously.

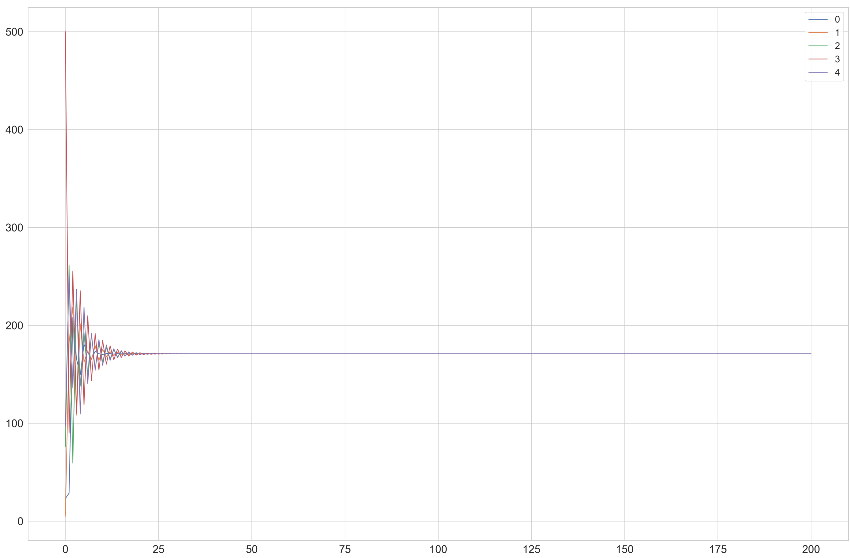
1.2 Opinion dynamics

In the second part of the first exercise the Λ matrix will be interpreted as the weight matrix of a graph. What is required now is to simulate the French-DeGroot dynamics on the previous graph with an arbitrary initial condition $x(0)$. Regardless of initial conditions a consensus status is reached immediately after few iterations, this is because the graph is strongly connected and aperiodic. The consensus value reached is equal to

$$\alpha = \sum_k \pi_k x(0)_k$$

that is the weighted average of the nodes' initial opinions. In Figure 2 is represented an example in which x(0) = [23, 5, 76, 500, 97], as can be seen the consensus is reached after about 20 epochs.

Figure 2: Convergence to consensus with 200 iterations



Now consider a random variable to initialize the x(0), that returns values in the range $[0, 1]$ and then has an average of $\frac{1}{2}$ and a variance of $\frac{1}{12}$, 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. To calculate the variance theoretically, the following formula has been used:

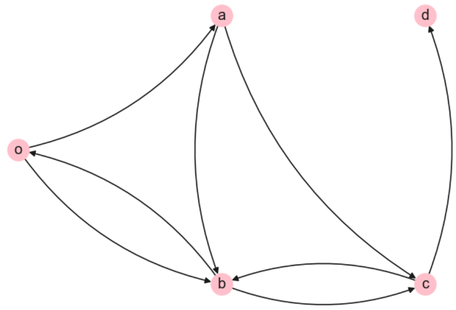
$$Var(\alpha) = Var(\sum_k \pi_k x(0)_k) = \frac{\sum_k \pi_k^2}{12} = 0.0170$$

While to calculate the variance by simulation several experiments have been performed and at each time, convergence of the final consensus has been calculated as follows

$$Var(\alpha) = \mathbb{E}[(\alpha - \mathbb{E}[\alpha])^2] = 0.0175$$

Removing the edges *(d, a)* and *(d, c)* the chart in Figure 3 is obtained Removing these

Figure 3: Graph once removed edges *(d,a)*, *(d,c)*

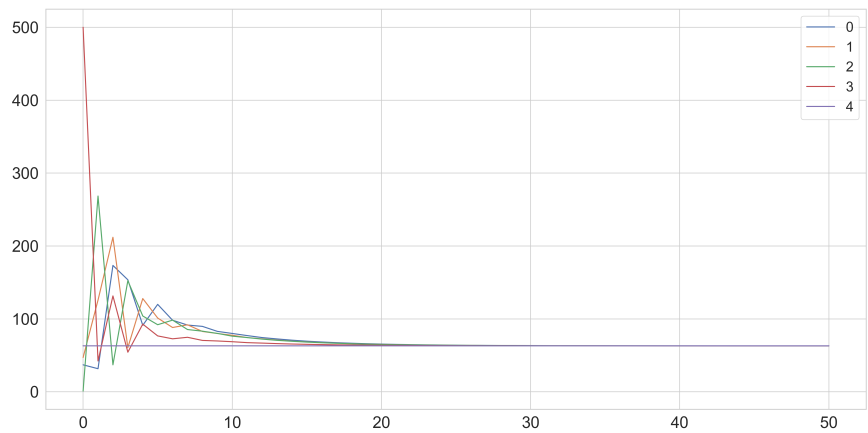


two links the starting graph is no longer strongly connected, but since in the graph there is a single sink (i.e a globally reachable connected component *C0*) which in our case is represented only by d than:

$$\lim_{t \to +\infty} x_i(t) = \pi' x(0), \forall i$$

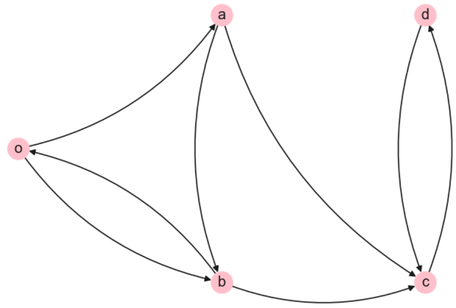
And also the opinions of nodes not belonging to the sink have no influence on the final consensus value, so the value of the consensus will depend only on d. Trying to simulate the dynamics with a certain condition, the value converge to the initial condition value of d, as can be seen in Figure 7 where the initial condition is equal to [37, 47, 1, 500, 63]. The variance of the consensus value has been calculated as before and the result obtained is

Figure 4: Convergence of dynamics starting with x(0)=[37, 47, 1, 500, 63]



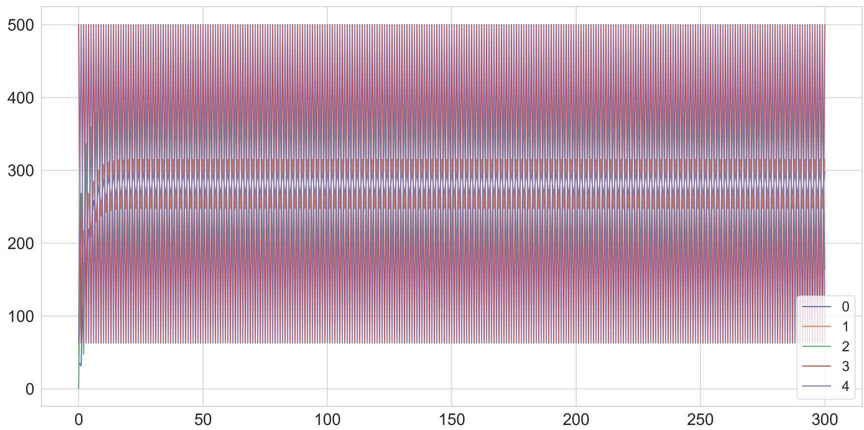
0.0871, as can be seen the value is much closer to that of the variance of the single uniform variable ($\frac{1}{12}$) this is due to the fact that the consensus converges to the value of the single variable linked to node d. While in the previous exercise it was far from this value because the value of the variance of the consensus was also influenced by the other nodes. Removing

Figure 5: Graph once removed edges *(d,a)*, *(c,b)*



the edges *(c,b)* and *(d,a)* the new graph represented in Figure 5 is obtained. Now as you can see from Figure 6 the dynamics never converge, this is because even if present only one sink component is periodic. In fact printing the values of the consent of each node after the 300 iterations is obtained x(300): [298.3, 315.1 163.8, 500, 63] where the initial condition is x(0) = [37, 47, 1, 500, 63]. As can be seen the opinion of the two nodes inside the sink is not affected by that of the nodes outside. The consensus is reached only if the two nodes in the sink component have the same opinion, otherwise consensus is not reached.

Figure 6:



2 Exercise

2.1 Simulation of many particles moving around in the network in continuous time

In this second exercise, it is necessary to again consider the graph in (Figure 1) and Λ as the transition matrix of a Continuous Markov Chain. In this case it is necessary to simulate the behaviour of 100 particles moving within the graph. Time follows an exponential distribution and the next position will depend on the normalized transition rate matrix $P = D^{-1} * \Lambda$

2.2 Particle prospective

Considering the particle prospective scenario, by starting 100 particles from node o and simulating for 60 time units with a global clock the same result as in 1.a is obtained. It is therefore possible to deduce that the particles have a motion independent of each other and it would have been sufficient to simulate 100 times the experiment produced in 1.a.

2.3 Node prospective

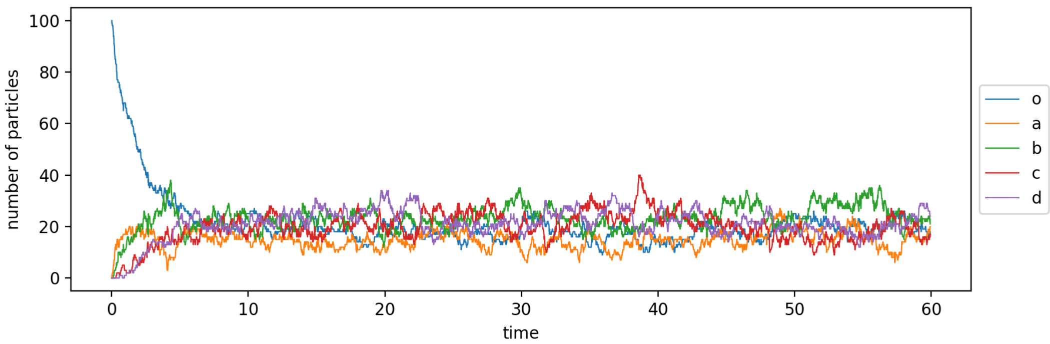
In this scenario, 100 particles start at node o, and the system is simulated for 60 time units. In this case we are not interested in the motion of the particles but only in the number of particles within each node at each instant of time. The Poisson clock rate is considered proportional to the number of particles within the node

Running the simulation 100 times, the distribution of the average particle number is as follows:

$$o : 19.23, a : 14.96, b : 21.61, c : 23.24, d : 21.96$$

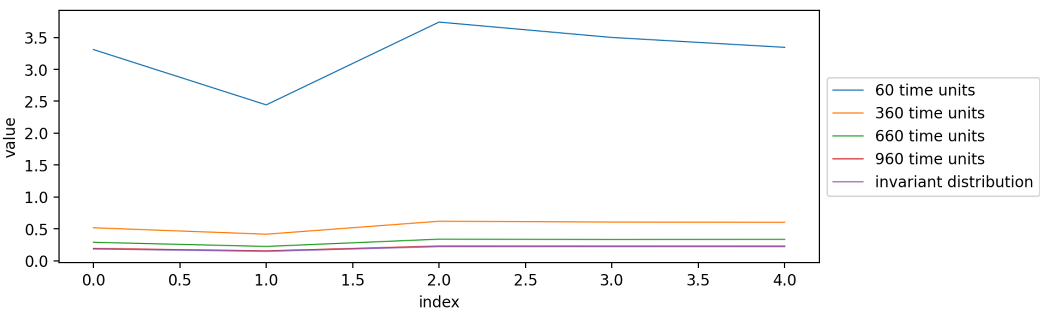
If we put 100 particles in node o and we wait for 60-time units, the number of particles per node are, in function of time, as in 7.

Figure 7: Graph representing the number of particles per node in function of time



The empirical frequencies are the fractions of total walk time that each node of the graph is visited in the random walk. In order to use empirical frequencies to approximate the invariant distribution of the graph, a random walk is constructed. the greater the number of iterations made, the more accurate the approximation will be, as is shown in 7.

Figure 8: Graph representing the convergence to the invariant distribution



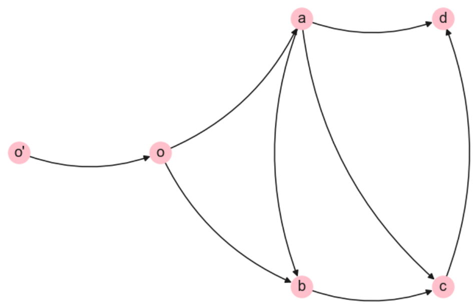
3 Exercise

In this part we will consider the network represented in Figure 9. A node o' has been added before o from which particles pass before entering the system. Also in the transition matrix Λ_{open} (3) a self loop was added on node d to avoid the all-zeros row

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

The number of particles in o' is not considered in the total number of particles in the system,

Figure 9: Graph representing the new network

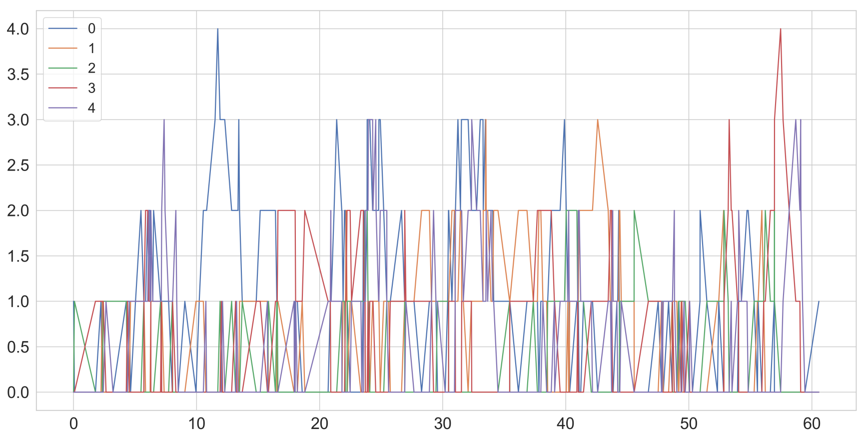


moreover the node d is considered as an exit node, essentially if within the algorithm the starting node is d then the particle is considered as leaving the system. In both scenarios the particles will enter o according to a Poisson process with rate 1.

3.1 Proportional rate

In this first scenario the particles will pass to the other nodes with a Poisson process with rate proportional to the number of particles present in the node. The system simulation in

Figure 10:



60 units of time, with an *input_rate* of 1, is shown in Figure 10

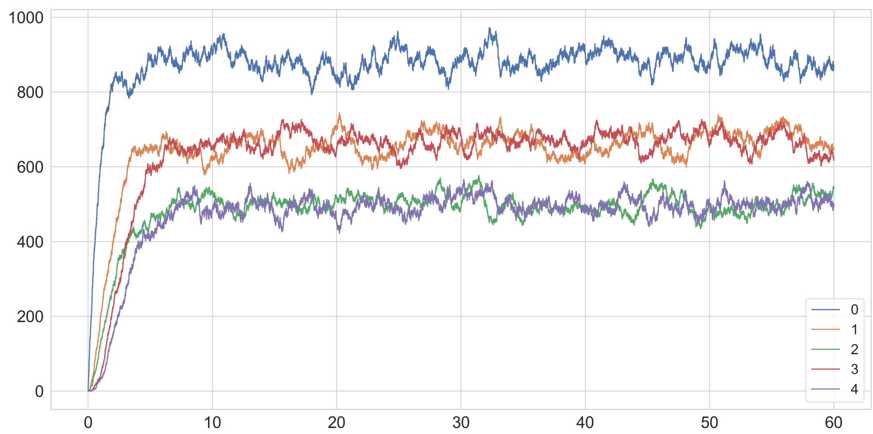
t_next is calculated as follows:

$$t\_next = \frac{-ln(u)}{n_{particles}[i] * \omega_i}$$

in the case of nodes without particles t_next has been considered infinite, while for o' it is equal to $\frac{-ln(u)}{input\_rate}$. In the simulation all the t_next associated to each node were calculated and the node with t_next minimum as the starting node was selected, while the arrival node was calculated through the matrix Λ_{open}.

To find the largest $input\_rate$ for which the system blowing up several tests were made with different input rates but with none of these the system exploded, as can be seen in Figure 11. This is due to the fact that the nodes 'adapt their velocity' according to that of the incoming particles.

Figure 11: system simulation in 60 units of time, with an $input\_rate$ of 1000



3.2 Fixed rate

In this second scenario the same experiments as before were simulated, the only difference is that each node i will pass along particles with a fixed rate ω_i. In Figure 12 the simulation of the system on 60 units of time and with input rate equal to 1 can be seen.

As you can see if the input rate is 1 the system is able to manage the flow of particles but increasing to 2 the $input\_rate$, as you can see in Figure 13, the system blowing up. Since the selected node from which the particles depart is the node with the minor t_{next}, and since the t_next of o' is equal to $\frac{-ln(u)}{input_{rate}}$ while that of o is equal to $\frac{-ln(u)}{1.125}$, if the input rate over the value of 1.125 the network is no longer able to handle the incoming particles.

Figure 12:

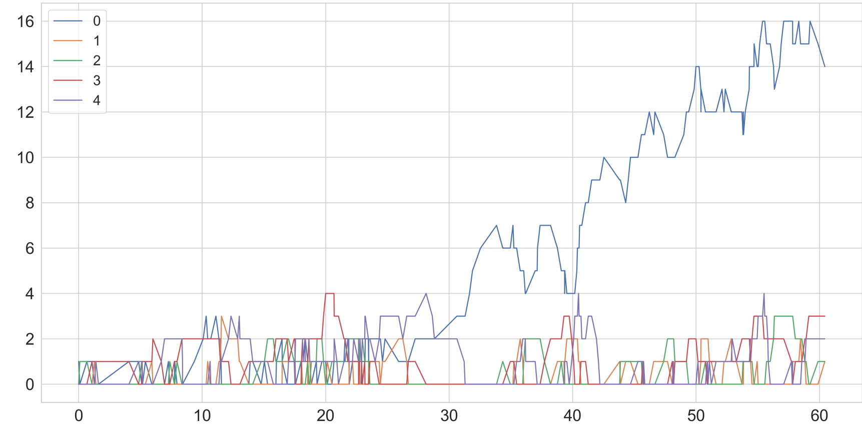


Figure 13: