Homework 2

Pierluigi Compagnone s288301
Armando La Rocca s279401
Network Dynamics and Learning
2021/2022

December 12, 2021

Collaboration with Giuseppe Desiderio

1 Exercise 1

The first exercise focuses on the behavior of a single particle moving in the a graph $\mathcal{G}$, showed in Fig. 1. The movement is modelled as a continuous-time random walk $X(t)$ associated with a transition rate matrix Λ showed in Fig. 2.

In the notebook *hw-2-2021-es1.ipynb* the code used to solve the proposed questions is provided.

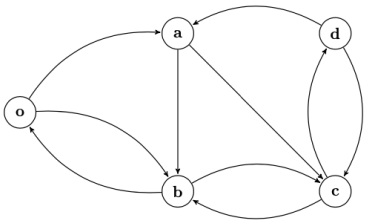

Figure 1: Graph $\mathcal{G}$

$$\Lambda = \begin{array}{c} \begin{array}{ccccc} o & a & b & c & d \end{array} \\ \begin{pmatrix} 0 & 2/5 & 1/5 & 0 & 0 \\ 0 & 0 & 3/4 & 1/4 & 0 \\ 1/2 & 0 & 0 & 1/2 & 0 \\ 0 & 0 & 1/3 & 0 & 2/3 \\ 0 & 1/3 & 0 & 1/3 & 0 \end{pmatrix} \end{array} \begin{array}{c} o \\ a \\ b \\ c \\ d \end{array} .$$

Figure 2: Transition rate matrix

First of all to deal with continuous-time Markov process there is one fundamental thing to define that is the Poisson clock, because the transitions happen at random time instants that are decided by the tick of this clock. A Poisson clock 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. To simulate a $rate - r$ Poisson process, one can simulate the time t_{next} between two ticks of the Poisson clock, which is a random variable with $rate - r$ exponential distribution. In fact, if drawing a uniformly distributed random variable $u \sim \mathcal{U}(0,1)$, t_{next} can be computed as:

$$t_{next} = -\frac{\ln(u)}{r}$$

where u is a random variable with uniform distribution, $u \in \mathcal{U}(0,1)$.

The process has been modeled according to a Global Poisson clock

Starting from node i, when *the global clock ticks* either the particle jumps to a neighbor j with probability $\overline{P}_{ij} = \frac{\Lambda_{ij}}{\omega_*},\ i \neq j$ or it stays in the same node (no transition) with probability $\overline{P}_{ii} = 1 - \sum_{i \neq j} \overline{P}_{ij}$

In this approach, the continuous time is "discretized" using a global clock, while the matrix $\overline{P}$ describes the jumps.

The theoretical return time, in notation T_j^+, is the minimum among all times such that a walk $x(t)$ ends where it started (node j).

$$T_j^+ = inf\{t \geq 1; X(t) = j\}$$

Considering an experimental approach, since the process is random, at the end of each simulation a different value of T_j^+ is found, so the average of this times is computed. The number of simulations done is 100000.

Looking at the graph $\mathcal{G}$ some properties can be exploited, in order to derive an analytical expression to calculate this average. In particular $\mathcal{G}$ is strongly connected, this implies a unique invariant distribution $\overline{\pi}$.

The expected return time $E_i[T_i^+]$ is theroretically defined as:

$$E_i[T_i^+] = \frac{1}{\omega_i \overline{\pi}_i} \quad \forall i \in \mathcal{V}$$

In particular considering as origin and destination the node a the result is the following one :

$$E_a[T_a^+] = 6.75$$

In the end the simulated result is in line with the theoretical one and the Squared Error is around 10^{-4}.

The expected hitting time of a Markov chain is the average time it takes to move from a state i to another state j.

$$T_i = inf\{t \geq 0; X(t) = j\}$$

To compute the average hitting time from o to d, the same strategy of the previous point is adopted. 100000 simulation are performed in order to estimate the average.

An analytical expression for the expected hitting-time $E_i[T_j]$ can be derived by the theory.

Let $X(t)$ be a Markov chain with finite state space $\mathcal{X}$, transition probability matrix P, and initial state $i \in \mathcal{X}$. Let $\mathcal{S} \subseteq \mathcal{X}$ be a subset of states that is reachable from every state $i \in \mathcal{X}$. Then, the expected hitting time $E_i[T_S]_{i \in \mathcal{X}}$ are the unique solution of the following linear system:

$$
\begin{aligned}
&E_i[T_S] = 0 \quad \text{if} \quad i \in \mathcal{S} \\
&E_i[T_S] = \frac{1}{\omega_i} + \sum_{j \in \mathcal{X}} P_{i,j} E_j[T_S] \quad \text{if} \quad i \notin \mathcal{S}
\end{aligned} \tag{1}
$$

where $P = diag(\omega)^{-1}\Lambda$.

Starting from this proposition the following system is constructed (computed by hand and here reported) according with $origin = o$ and $destination = d$, meaning that in the set $\mathcal{S}$ there is just the node d.

$$
\begin{cases}
E_o[T_d] = \frac{1}{\omega_o} + P_{oa}E_a[T_d] + P_{ob}E_b[T_d] \\
E_a[T_d] = \frac{1}{\omega_a} + P_{ab}E_b[T_d] + P_{ac}E_c[T_d] \\
E_b[T_d] = \frac{1}{\omega_b} + P_{bo}E_o[T_d] + P_{bc}E_c[T_d] \\
E_c[T_d] = \frac{1}{\omega_c} + P_{cb}E_b[T_d]
\end{cases}
$$

Here is reported the result obtained solving the proposed formulation on the notebook.

$$E_o[T_d] = 8.75$$

Comparing the theoretical result and the one obtained by simulations the error obtained is around 10^{-3}.

Let $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{W})$ be a weighted graph, and $x(t) \in \mathrm{R}^V$ denote the state of the nodes of the graph The dynamics of $x(t)$ is described by the following equation

$$x(t+1) = Px(t) \quad t = 0, 1, ... \tag{2}$$

where P is the normalized adjacency matrix, and $x(t) = (x_i(t))_{i \in \mathcal{V}}$ is the vector of all agents' opinions. The consensus is a particular configuration where all the agents converges to the same opinion in a certain amount of time.
However the consensus exists only if the following conditions are satisfied (Theorem Asymptotics of French-De Groot model) :

1. Condensation graph has 1 sink because it must have a single trapping set.

2. The sink component of the graph is aperiodic because periodicity does not allow a proper mixing of the opinions, indeed simulating the dynamic a never ending oscillations during time is observed.

If the two condition are satisfied then,

$$\lim_{t\to+\infty} x(t) = \alpha\mathbf{1},$$

i.e., the agents get to consensus $\alpha = \pi' x(0)$, where $\pi = P'\pi$.

Since the Graph $\mathcal{G}$ satisfy the presented conditions a consensus value will be reached. In particular choosing the following initial conditions:

$$x(0) = [10, 11, 12, 13, 14]$$

the nodes' opinion converge to a consensus $\alpha = 12.17$. The evolution of the opinions is represented in Fig. 3, which shows how the convergence to the consensus is reached in less than 20 iterations.

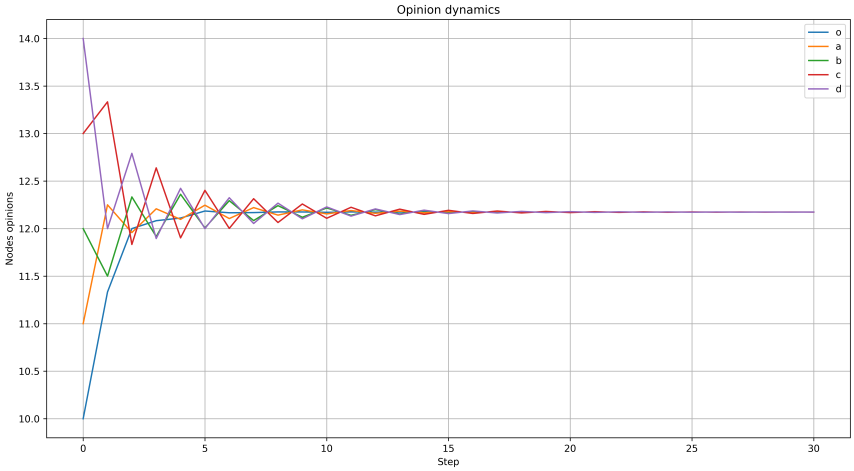

Figure 3: Opinions evolution on graph $\mathcal{G}$

Assuming now the initial condition given by $x_i(0) = \xi_i$ where $\{\xi_i\}_{i\in V}$ are i.i.d random variables drawn by a normal distribution with variance $\sigma^2 = 25$ and mean $\mu = 0$.

The consensus Variance is given by the following formula (application of wisdom of crowds):

$$Var(\alpha) = \sigma^2 \sum_k \pi_k^2 = 5.3 \tag{3}$$

A similar result is achieved with the simulation, evaluating the consensus values reached with different initial conditions $x(0)$ sampled from the same distribution and computing its variance. The quadratic error considering 100000 repetitions accounts around 10^{-4}.

Removing the edges (d, a) and (d, c) from $\mathcal{G}$ the graph $\mathcal{G}_g$ is obtained and showed in Fig. 4.

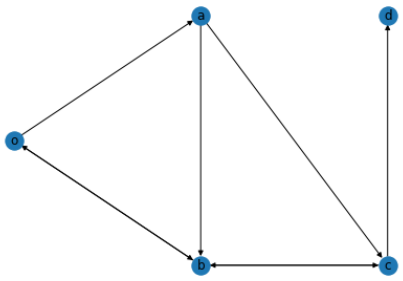

Figure 4: Graph $\mathcal{G}_g$

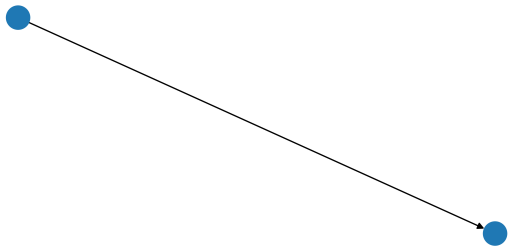

Figure 5: Condensation graph associated to $\mathcal{G}_g$

In Fig. 5 the condensation graph for $\mathcal{G}_g$ is represented which has one sink (composed by node d). Also for $\mathcal{G}_g$ the dynamics will converge to an equilibrium, since the French-DeGroot conditions are satisfied. In contrast with the previous point, where the consensus is a sort of average of the opinions in the different nodes, here the consensus is the initial opinion of the sink component, the rest of the nodes do not influence the dynamic. In this case node d is called **stubborn** since doesn't change its opinion.
Furthermore, in a very informal way, it can be said that it will "convince" all the other nodes to agree with its opinion. Anyway, the consensus exists and is given by the initial opinion of the sink. In Fig. 6 is showed the evolution of the dynamic.

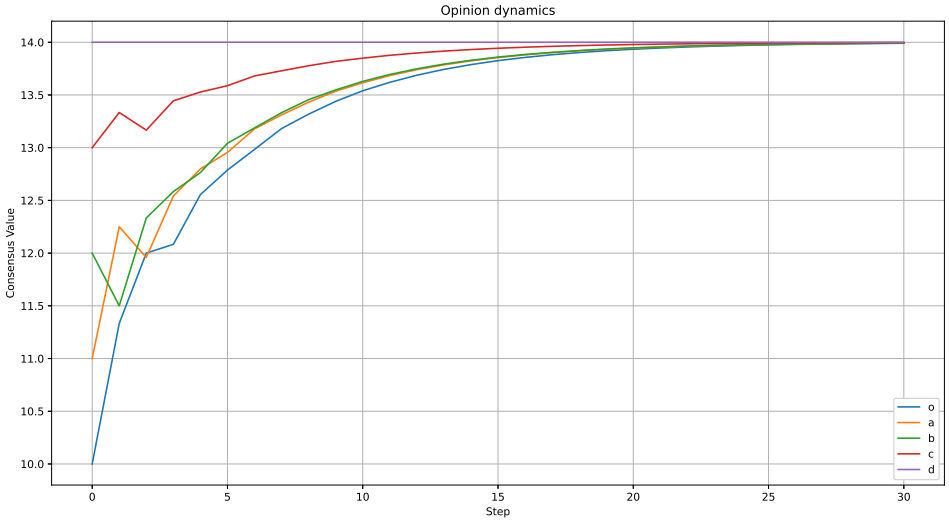

Figure 6: Opinions evolution in $\mathcal{G}_g$

To compute the variance of the consensus, the initial conditions $x_i(0) = \xi_i$ are drawn from a normal distribution with $\sigma^2 = 25$, then after computing the invariant distribution for the new graph $\mathcal{G}_g$ the same formula as in Eq.(3) has been used to calculate the variance.

$$Var(\alpha) = \sigma^2 \sum_k \pi_k^2 = \sigma^2$$

The consensus variance remains σ^2 because the structure of the invariant distribution π. From theory is known that the invariant distribution has support only on the sink components, here there is a unique sink (node d), indeed the invariant distribution has shape $\pi = [0, 0, 0, 0, 1]$, this implies the result found above. Moreover the results is confirmed by experiments and the quadratic error between simulated and theoretical variance is $2 \cdot 10^{-3}$.

Removing the edges (c, b) and (d, a) the graph $\mathcal{G}_h$ in Fig.7 is obtained.

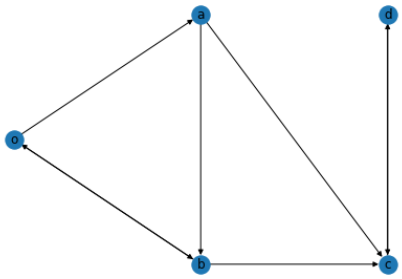

Figure 7: Graph $\mathcal{G}_h$

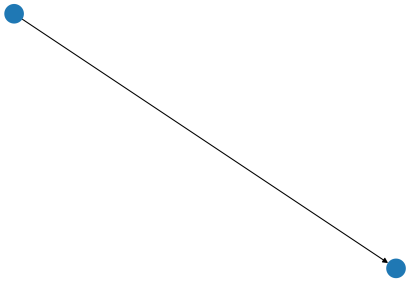

Figure 8: Condensation graph for $\mathcal{G}_h$

The graph $\mathcal{G}_h$ has one sink in the condensation graph, as showed in Fig. 8. The difference now is that such component is not aperiodic, in fact the sink is composed of nodes c and d which are linked by a bidirectional connection, which imply a period of 2, indeed, the dynamic does not converge to a consensus state. As showed in Fig. 9 the opinions oscillates between the ones of node d and node c .

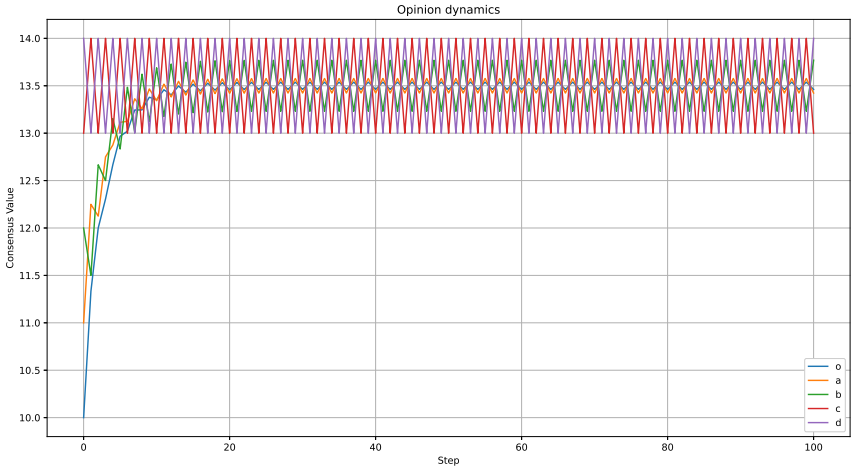

Figure 9: Opinions evolution of $\mathcal{G}_h$

2 Exercise 2

The graph $\mathcal{G}$ is the same of Exercise 1, whose structure and transition rate matrix are respectively represented in Fig. 1 and Fig. 2. However, now the proposed task consists in simulating the random walk of many particles around the network in continuous time, from two different point of views:

- Particle perspective
- Node perspective

In the notebook *hw-2-2021-es2.ipynb* the code used to solve the proposed question is provided.

2.1 Particle perspective

Observing the dynamic from the particles standpoint is conceptually the same analysis performed at the beginning of the previous exercise, with the difference that now there are 100 particles in the network. To face this situation some changes are necessary, in particular the dynamic of each particle is tracked using an array to store the position of all the particles at each tick of the clock. Moreover a global Poisson clock is used to simulate the dynamic of all the units, so the rate of the clock is multiplied by the number of particles respect to the case of a single one, in order to replicate the movement of all the particles in parallel in the network at the original rate.
More in details, the question is about checking the average return time of a particle in node a, supposing that all the 100 particles start from the same position. To do this the particles' positions are initialized in a, then at each tick of the global clock the particle to move is randomly selected and its new position is determined according to the probabilities in matrix $\bar{P}$, defined in the previous exercise.
Then in order to analyze the return time there are two interpretation:

- Chose a specific particle to follow and stop when this one return to a;
- Follow all the particles and stop when the first one return to a.

The first approach presents a very similar result to the case with one particle for two main reasons: the first is that the Poisson clock is 100 times faster and the second one is that the behaviour of a particle is independent from the others. This statement is also proved choosing a random particle between the proposed 100 and taking the return time. This operation is performed 1000 times averaging the results and the obtained one is very near to the one obtained in the first exercise considering just one particle.

In the second approach, as soon as arrival position coincides with a, it means that the first particle is back, so the loop is stopped and the return time is stored. The dynamic is simulated 10000 times and the average return time obtained is 0,67.
The average return time is about ten times smaller of the single particle case. Considering this approach, the return time scales of a factor $\sqrt{\#particles}$, which can be considered a reasonable effect, because having multiple particles and choosing the first one to come back is like selecting a best case.
In the following Figure is reported a comparison between the square root of the number of particles and the ratio between the average return time with one particle on the average return time considering a different number of particles.

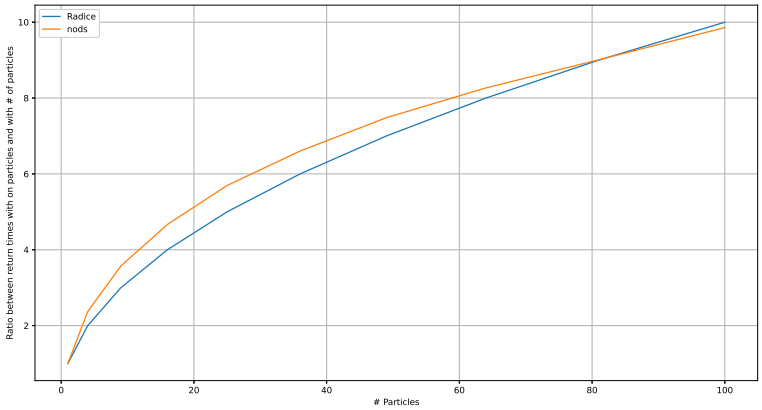

Figure 10: Comparison of $\sqrt{\#Particles}$ and the scaling factor of the average return time with one particle respect to the average return time with many particles

2.2 Node perspective

The proposed task is simulating the dynamics of 100 particles, which are initially in node o, for 60 time units and looking at the system from the nodes' point of view.
Thus, the number of particles in each node is stored instead of the position for each particle. Moreover, the nodes will pass along particles at a rate proportional to the their cardinality in the node. To deal with it, the selected strategy is to use a global clock with rate equal to the total number of particles. Then at every tick the node from which a particle starts is randomly selected, but proportionally to the number of particles in it. At the end, the arriving node is selected according to the probability in matrix $\bar{P}$ and the number of particles in each node after the transition is stored.
The dynamic is analyzed for 60 time units, and the average quantity of particles in nodes during the simulation is:

$$[18.08,\ 15.01,\ 22.36,\ 22.57,\ 21.99]$$

The trend of the number of particles in each node is shown in Figure 11.

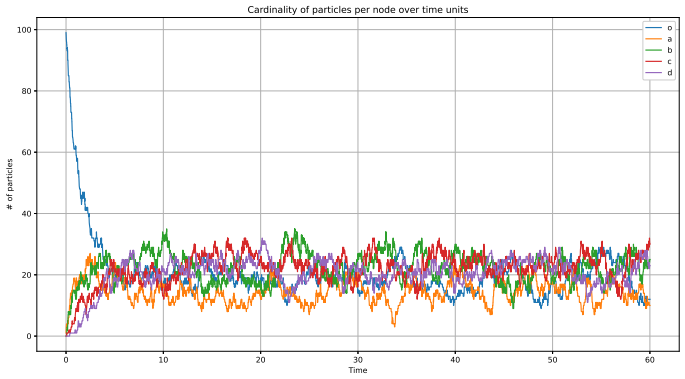

Figure 11: Number of particles in the nodes of $\mathcal{G}$ over the time

Since the Graph $\mathcal{G}$ is strongly connected and aperiodic the random walk X(t) is ergodic and the stationary distribution $\bar{\pi}$ ($\bar{\pi} = \bar{P}\bar{\pi}$) can be interpreted as the fraction of particles in each node when $t \to \infty$. This statement is also confirmed by the obtained results. In fact computing $\bar{\pi}$ the following result is obtained :

$$\bar{\pi} = [0.1851,\ 0.1481,\ 0.2222,\ 0.2222,\ 0.2222]$$

Considering that in the network there are 100 particles the interpretation of spatial average of $\bar{\pi}$ stated by the ergodic theorem is verified.

3 Exercise 3

Given the open Graph $\mathcal{G}$ in Figure 12 and the transition matrix Λ_{open} in Figure 13, the goal of this exercise is to study how different particles affect each other when moving around in a network in continuous time. Assuming that particles enter $\mathcal{G}$ through node o with a given rate λ and get out from node *d*.

In the notebook *hw-2-2021-es3.ipynb* the code used to solve the proposed question is provided.

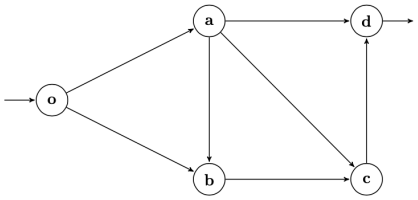

Figure 12: Graph G

$$\Lambda_{\text{open}} = \begin{array}{c} \begin{matrix} o & a & b & c & d \end{matrix} \\ \begin{pmatrix} 0 & 2/3 & 1/3 & 0 & 0 \\ 0 & 0 & 1/4 & 1/4 & 2/4 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix} \end{array} \begin{matrix} o \\ a \\ b \\ c \\ d \end{matrix}$$

Figure 13: Transition rate matrix

In order to analyze the particles behaviours two different scenarios are considered where the main difference is given by the rate nodes pass along particles.

- Proportional Rate
- Fixed Rate

In both the approaches, in order to manage the process of movement inside the network each node is associated to a local clock. Instead, an input clock regulates the entry of the particles inside the network. At each step, the transition depends on the first ticking clock, in particular if the input clock ticks as first a particle will enter the system. While, if the first ticking clock is related to a node, a particle will move from it to a new position, accordingly to the transition probabilities of the network. Except for node d, because its clock is associated to the exit of a particle from $\mathcal{G}$.
Of particular interest is the largest input rate that the network can bear without blowing up. Intuitively a system can handle a certain input rate if it evolves without a dramatic accumulation of particles in its nodes over the iterations. For this purpose a consideration can be made about the

network's structure, indeed a particle that enters in o will reach d in a finite time, without any possibility to get stuck in other nodes. For this reason, the capacity of the graph to handle a high input rate without blowing up is mainly influenced by the ratio between the input rate λ and the output rate λ_d.
In particular, the system explodes if:

$$\frac{\lambda}{\lambda_d} > 1 \tag{4}$$

Moreover, the magnitude of this ratio is proportional to the time the system takes to blow up.
In both the proposed scenario the dynamic is analyzed for 60 time units and different values of input rate λ are considered.

3.1 Proportional Rate

Considering a proportional rate the departure times of a node can be seen as a Poisson process with rate $n_i(t)\omega_i$, where t is the current time, $n_i(t)$ the number of particles in node i at time t and $\omega = \Lambda_{open}1$. Instead the input rate will be equal to λ. Intuitively, if a node has a very high number of particle there will be an higher probability that a particle will move from that node to one of its neighbors. This behaviour avoids the accumulation of particles on a single node, hence the system can handle an arbitrary input rate λ. In fact λ_d changes over the iterations based on the number of particles inside the system, balancing the rate of input, so the system will never explode. In the following figures are represented the results obtained with $\lambda = 1$ and $\lambda = 100$, in order to verify the previous statement.

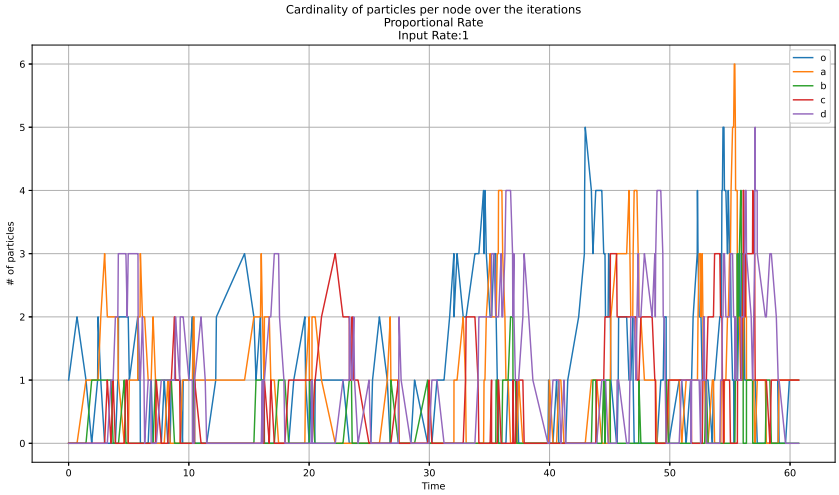

Figure 14: Cardinality of particles in node of $\mathcal{G}$ considering proportional scenario with $\lambda = 1$

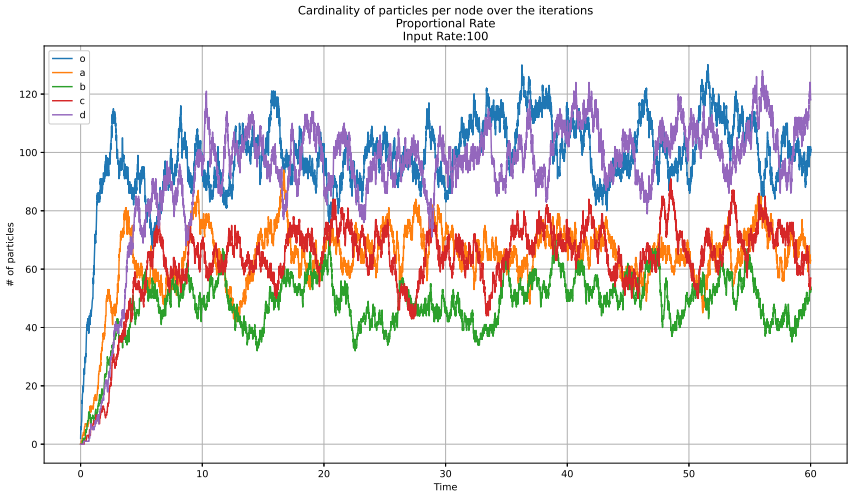

Figure 15: Cardinality of particles in node of $\mathcal{G}$ considering proportional scenario with with $\lambda = 100$

3.2 Fixed Rate

In this case the clocks of the nodes have a fixed rate equal to 1. From a practical point of view, now it could happens that the first clock to tick is associated to an empty node, because the rates are not linked anymore to the cardinality of particles on the correspondent nodes. Obviously this situation have to be avoided and it is done in implementation phase. The following chart shows the trend of nodes' cardinality with an input rate 1.

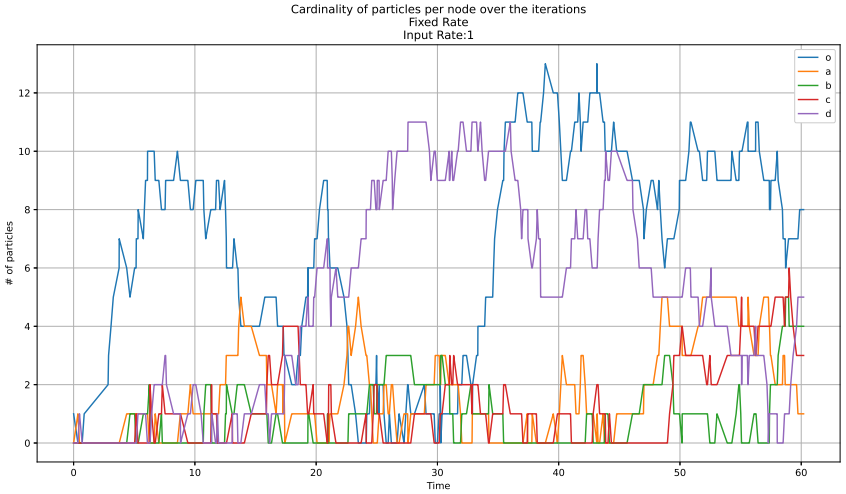

Figure 16: Cardinality of particles in node of $\mathcal{G}$ considering fixed scenario with $\lambda = 1$

Taking as reference the equation 4, now the maximum input rate that the network can handle without exploding is 1. This is a consequence of the fact that the output rate is fixed to 1, so for every input rate greater than 1 the inequality 4 would be satisfied. Moreover, as higher is this ratio as faster the dynamic blows up.
In Figure 17 is shown the cardinality of the nodes with 1.1 as input rate, just to stress the fact that the system explodes as soon as the rate is greater than 1, even if it takes a while for values slightly greater than the threshold.

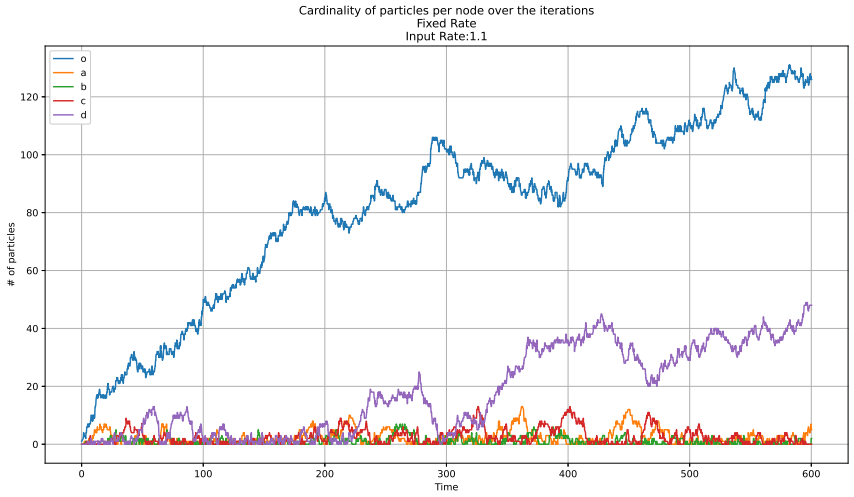

Figure 17: Cardinality of particles in node of $\mathcal{G}$ considering fixed scenario with $\lambda = 1.1$