Politecnico di Torino

Networks Dynamics and Learning

Homework #2

Antonio Albanese, 282043

December 2021

In this homework I exchanged ideas and compared results with my colleague Nedescu Ionut Cosmin.

Figures and equations from the assignment

Problem 1.

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

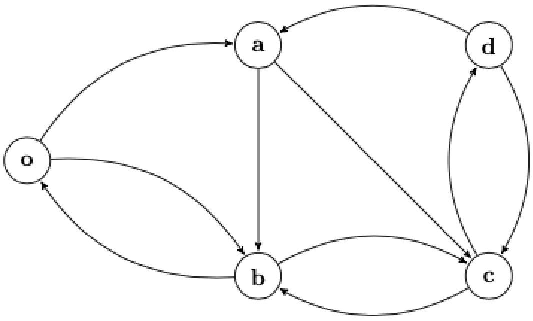


Figure 1: Closed network in which particles move according to the transition rate matrix (1)

Problem 3.

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

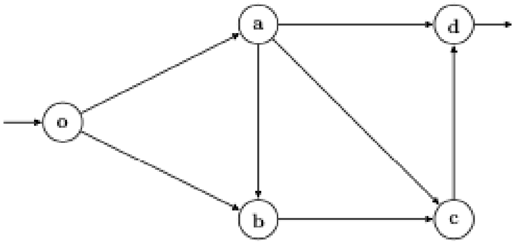


Figure 2: Open network according to the transition matrix in (2)

Problem 1 - Answer

In the first part of this problem we are focusing on random walk of a particle in a weighted graph based on the transition probability Λ, basically we are talking about Markov chain in continuous time.

Definition 1 (Random walk) *A random walk on a graph is a process that begins at some node, and at each time step moves to another node.*

Definition 2 (Markov chain) *A stochastic process X(t) is said to be a* ***Markov chain*** *if satisfies the* ***Markov property****: the dependence of the future state on the history is limited to the present state.*

Definition 3 (Continuous-time Markov Chain)

- *Given the Poisson process*

$$N_t := \sup\{k \geq 0 : T_k \leq t\}, \quad t \geq 0; \tag{3}$$

- *Given a finite state space $\mathcal{X}$ and a square nonnegative matrix $\Lambda = (\Lambda_{ij})_{i,j\in\mathcal{X}} \neq 0$ with entries indexed by elements of $\mathcal{X}$ and diagonal entries $\Lambda_{ii} = 0$*

- *Let:*

$$w_i = \sum_{j\neq i} \Lambda_{ij}, \quad i \in \mathcal{X}, \quad w_* = \max_{i\in\mathcal{X}} w_i \tag{4}$$

- *Consider a rate-w_* Poisson clock $T0 \geq T1 \geq ...$, the associated Poisson process N_t, and an independent discrete-time Markov chain $U(k), fork = 0, 1, ...,$ with transition probabilities*

$$\overline{P}_{ij} = \frac{\Lambda_{ij}}{w_*}, \quad i \in \mathcal{X}, \quad \overline{P}_{ii} = \sum_{j\neq i} \overline{P}_{ij} \tag{5}$$

  The ***discrete time Markov chain*** *is*

$$X(t) = U(N_t), \qquad t \geq 0 \tag{6}$$

  where N_t is called the jump chain associated to the continuous-time Markov chain $X(t)$.

To accomplish the required task the `SimulateParticle` function has been built. That function will be used to produce simulations for both *return time* and *hitting times* estimation.

(a) The average time obtained varies depending on the number of simulations computed, and it tends to the theoretical value computed at the point *b)* as the number of simulations tends to infinite. Some results obtained are in Table (1).

# of simulations	**Avg return time**
1000	6.6865
10000	6.7037
100000	6.7362

Table 1: Average return time on different number of simulations

(b) At this point we need to remark that the given grpah is **strongly connected**, and also we need to enunciate the following theorem

Theorem 1 *Let $X(t)$ be a continuous-time Markov chain with finite state space $\mathcal{X}$ and transition rates $\Lambda_{ij} fori \neq j \in \mathcal{X}$. Let Λ be the nonnegative square matrix with out-diagonal entries equal to the transition rates and diagonal entries $\Lambda_{ii} = 0$. Let $\mathcal{G}_\Lambda = (\mathcal{X}; \mathcal{E}; \Lambda)$ be the graph with node set $\mathcal{X}$, and weight matrix Λ, and let $w = \Lambda\mathbb{1}$, $P = diag(w)^1\Lambda$ be its degree vector and normalized weight matrix, respectively. If $\mathcal{G}_\Lambda$ is strongly connected, then*

...

(iv) the expected return times satisfies

$$\mathbb{E}_i[\overline{T}_i^+] = \frac{1}{w_i\overline{\pi}_i} \tag{7}$$

On the other hand, if $\mathcal{S} \in \mathcal{X}$ is a subset of states that is globally reachable in $\mathcal{G}_\Lambda$, then

(v) the expected hitting times $\mathbb{E}_i[T_{\mathcal{S}}]$ are the unique solution of

$$\mathbb{E}_{\mathcal{S}}[T_{\mathcal{S}}] = 0, \qquad \mathbb{E}_i[T_{\mathcal{S}}] = \frac{1}{w_i} + \sum_{j\in\mathcal{X}} \overline{P}_{ij}\mathbb{E}_j[T_{\mathcal{S}}], \qquad i \in \mathcal{X}/\mathcal{S} \tag{8}$$

...

Calculating the expected return time on node a as state in Theorem (1) Equation (7) gives as result $\mathbb{E}_\mathbf{i}[\overline{\mathbf{T}}_\mathbf{i}^+] = \mathbf{6.7500}$.

As already said, if the number of simulations in point *a)* tends to infinite their result will be closer to $\mathbb{E}_i[\overline{T}_i^+]$.

(c) Again, the average hitting time obtained varies depending on the number of simulations computed, and it tends to the theoretical value explained at the point *d)* as the number of simulations tends to infinite. Some results obtained are in Table (2).

# of simulations	**Avg hitting time**
1000	9.0984
10000	8.8634
100000	8.7432

Table 2: Average hitting time of a particle that moves from node o to node d on different number of simulations

(d) Hitting times can be computed as stated in Theorem (1) Equation (8). The result obtained with the given graph and transition matrix, for a random walk from node o to node d is $\mathbb{E}_o[T_d] = \mathbf{8.7857}$.

(e) Simulating the system as required we can see that the dynamics converge to a consensus. To explain that, we need some definitions.

Definition 4 (French - De Groot opinion dynamics) .
Consider a network described as a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, W)$, where the node set $\mathcal{V}$ represents a population of agents, the link set $\mathcal{E}$ represents interactions among agents, and the entries W_{ij} of the weight matrix quantify the strength of the influence that agent j has on agent i. Let P be the normalized weight matrix of $\mathcal{G}$. Assume that every agent i has a state $x_i(t) \in \mathbb{R}$ that is updated at discrete time steps $t = 0, 1, ...$ in response to the current states of her/his out-neighbors in $\mathcal{G}$ according to the following linear averaging rule

$$x(t+1) = Px(t), \qquad t = 0, 1, ... \tag{9}$$

where $x(t) = (x_i(t))_{i\in\mathcal{V}}$ is the vector of all agents' opinions. Equation (9) is the ***linear averaging dynamics on*** $\mathcal{G}$. *In social science Equation (9) is known as the* ***French-DeGroot opinion dynamics model****.*

Proposition 1 *Given a graph $\mathcal{G} = (\mathcal{V}; \mathcal{E}; W)$ and $P = D^{-1}W$, consider the linear averaging dynamics (9). The following facts hold true:*

...

(ii) For every initial condition x_0, the evolution $x(t)$ satisfies

$$\pi' x(t) = \pi' x_0 \tag{10}$$

for every π invariant distribution of P.

Suppose now that for an initial condition $x_0 \in \mathcal{R}^{\mathcal{V}}$ the evolution $x(t)$ determined by (9) converges to some $\overline{x} \in \Omega$. Then, the following facts hold true:

(iii)

$$P\overline{x} = \overline{x}; \qquad \pi' x_0 = \pi' \overline{x} \tag{11}$$

for every π invariant distribution of P.

(iv) If, in addition, $s_{\mathcal{G}} = 1$,

$$\overline{x} = \mathbb{1}\pi' x_0 \tag{12}$$

where π is the unique invariant distribution of P

Proposition 2 *Let $\mathcal{G}$ be a graph such that $s_{\mathcal{G}} = 1$ and the connected component of $\mathcal{G}$ corresponding to the unique sink of its condensation graph is aperiodic. Let P be the normalized weight matrix of $\mathcal{G}$ and let $\pi = P'\pi$ be its unique invariant probability. Then, the discrete-time averaging dynamics (9) satisfies*

$$\lim_{t\to+\infty} x(t) = \alpha\mathbb{1}, \qquad \alpha = \pi' x(0) \tag{13}$$

In the given graph there are all conditions for which, through Proposition (2), we can affirm that for sure the **dynamics will converge to a consensus**, which value can be found with Equation (13).

In the practice in the script the initial state of each node is randomly determined using the function `numpy.random.rand`, then it is possible to compute the evolution of the French - De Groot dynamics and compare it with the theoretical value, obtaining the same result. In Figure (3) we can see how the evolution of the simulated dynamics converge to the theoretical value of the consensus.

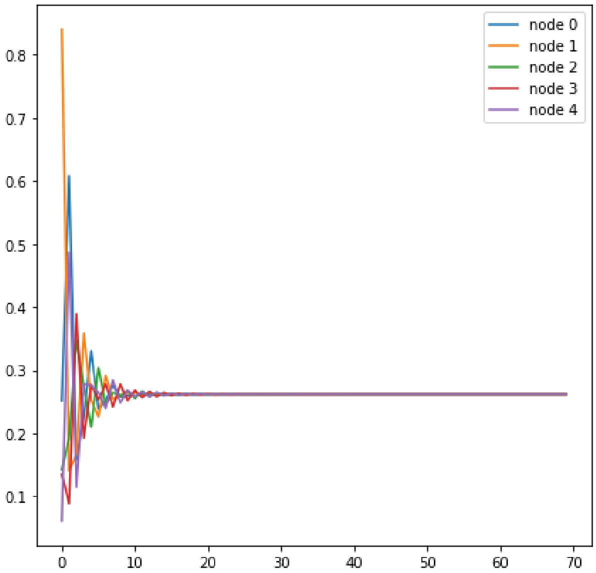


Figure 3: Convergence of the French - De Groot dyanmics

(f) We can assume that a community of agents is represented by the given graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, W)$ that is strongly connected and aperiodic and there is an underlying state of the world represented by a scalar parameter θ, and every agent $i \in \mathcal{V}$ observes a noisy version of it, that corresponds with its initial opinion $x_i(0) = \xi_i$.

$\{\xi_i\}_{i \in \mathcal{V}}$ are $i.i.d$ random variables, created in the practice with the function `numpy.random.rand` that return values uniformly distributed between 0 and 1, with $\mathbb{E}[\xi] = 0.5$ and $\sigma^2 = 1/12$. We can consider now the initial states as

$$x_i(0) = \xi_i = \theta + \psi_i$$

where ψ_i is the noise described above. From the general theory we know that

$$\overline{x} = \sum_{k \in \mathcal{G}} \pi_k x_k(0) = \theta + \sum_{k \in \mathcal{G}} \pi_k \psi_k$$

where π is the invariant distribution centrality of $\mathcal{G}$. $\overline{x}$ is thus a random variable whose expectation coincide with θ, while its variance is

$$\sigma_{\overline{x}}^2 = \sigma^2 \sum_{k \in \mathcal{G}} \pi_k^2 \tag{14}$$

The theoretical value of the variance obtained with Equation (14) is $\sigma_{\overline{x}}^2 = 0.0178$

Simulating the French - De Groot dynamics on the given system we obtain a variance $\sigma^2_{\bar{x},sim} = 0.0171$ that is similar to the theoretical value $\sigma^2_{\bar{x}}$.

Here it is also important to notice that the variance of the consensus is less than the variance of the initial opinion, meaning that the crowd is wiser than any single individual. This says that, as long as the graph describing the influence among individuals is connected and the individuals have all the same measurement capabilities (e.g. same error variance), the linear averaging dynamics always leads to an estimation of the true state θ that is strictly better than their original estimation.

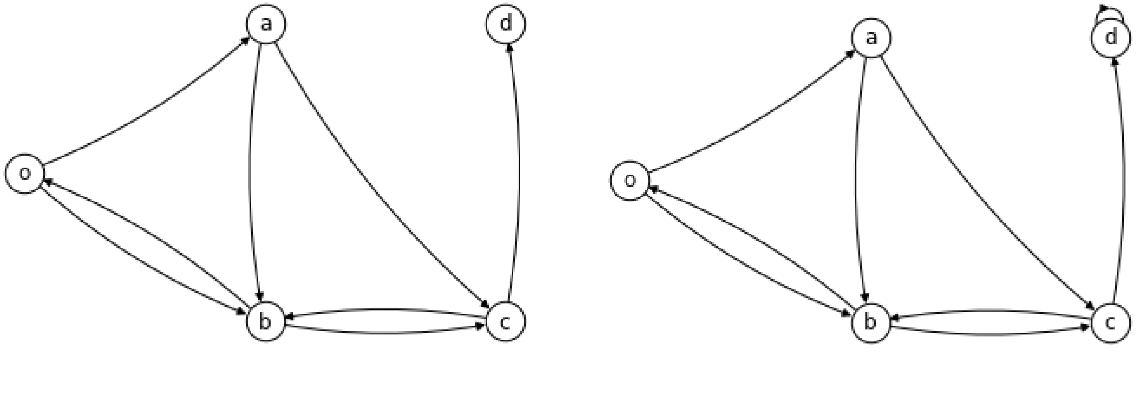
This case is known as **the wisdom of crowd**.

(g) Removing edges (d, a) and (d, c) the node d become a sink node as in Figure (4a), to simulate the French - De Groot dynamics we need to add a self-loop to the node d, as in Figure (4b). Doing so, the node d will not be influenced by other agents in the network, but will act as a *stubborn* node. In fact computing the invariant distribution π we obtain as expected $\pi = [0, 0, 0, 0, 1]$, that indicate that the consensus value computed as in Equation (13) will depend only from the initial opinion of the node d. Simulations computed sustain the previous assumption, as reported in Table (3).

Initial opinions	**Empirical consensus**	**Theoretical consensus**
$[10, 0, 0, 0, 1]$	1.0000	1
$[94, 43, 32, 1, 0]$	0.0000	0
[0.37030.58320.69850.24990.4909]	0.4909	0.4909

Table 3: Simulations with different initial states for the node d, initial opinions refers to nodes $[o, a, b, c, d]$

While, considering the initial opinions as random variables and computing the variance on the consensus value with 1000 simulations we obtain $\sigma^2_{\bar{x}} = 0.0825$, much more closer to the variance of the of the initial opinion vector. In fact at each simulation the the consensus will be the value of initial state of node d and that value is chosen among a uniform distribution $\mathcal{U}(0, 1)$ that has variance $\sigma^2 = 1/12$.



(a) Graph $\mathcal{G}$ with removed edges (d, a) and (d, c) (b) Graph $\mathcal{G}$ with self-loop added on node d

Figure 4

(h) Again we start considering the graph in Figure (1) and remove edges (c, b) and (d, a), and we obtain the graph in Figure (5)

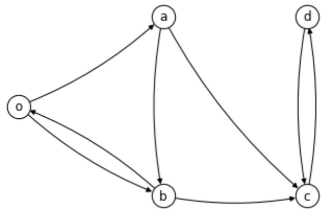


Figure 5: Graph obtained removing edges (c, b) and (d, a)

Randomly choosing the initial opinions, the French - De Groot dynamics **do not converge**. With a better look at the graph we can see that removing edges the graph is not connected anymore, in fact from the node d we can only reach node c and vice-versa. Now the graph does not satisfy the requirements to calculate the consensus value with Equation (13), but the corresponding condensation graph has still only 1 sink. In this situation the only way to make the system reach a consensus, is to **give to all the node in the sink component the same initial opinion**, in this way the sink component nodes will act together as a sink node ad the French - De Groot dynamic will converge to their initial opinion. Some simulations are reported in the Table (4) and Figure (6)

Initial opinions	**Empirical consensus**
$[0.4679, 0.3616, 0.2771, 1, 1]$	1
$[0.8525, 0.4262, 0.6882, 7, 7]$	7
$[0.1445, 0.7719, 0.3533, 4, 4]$	4
$[0.2118, 0.3410, 0.5384, 2, 2]$	2

Table 4: Simulations with different initial states for the node d and c, initial opinions refers to nodes $[o, a, b, c, d]$

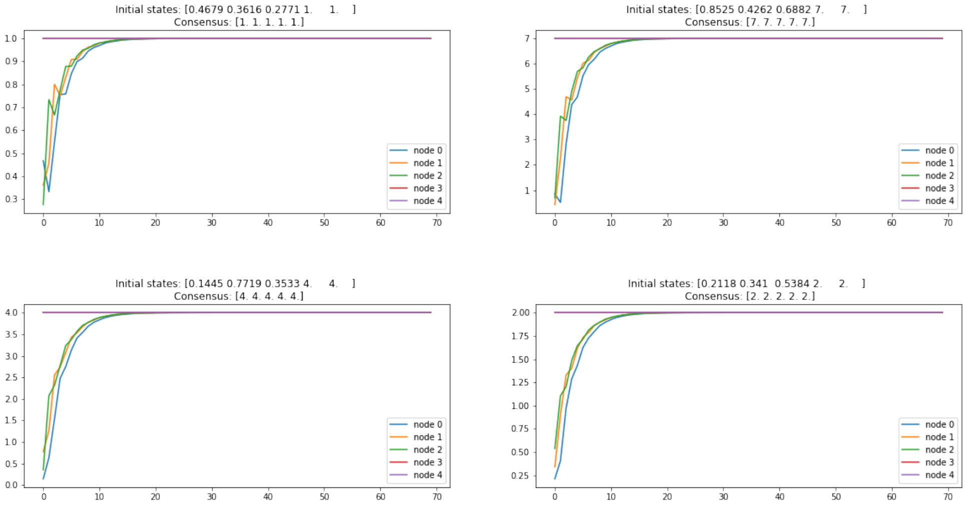


Figure 6: Convergence to a consensus when nodes c and d are at the same initial opinion

Problem 2

In this problem we are considering again the graph in figure (1) with weights according to (1).

(a) We want to observe the average return times of 100 particles starting from node a. To do so, the function `particlePerspectiveParticleClock` has been built. The average return time on the node a over 100 particles is

$$\overline{T}_a^+ = 6.7890$$

and we can see how close this value is to the theoretical return time found in *Problem 1 Point b* ($\mathbb{E}_i[\overline{T}_i^+] = 6.7500$).

(b) To answer the first question the function `NodePerspectiveGlobalClock` has been built, with this function we can simulate `nParticles` moving in the graph during `time` units of time for `nSimulations` simulations. At the end we can compute the average of the number of particles in each node at the end, across all simulations. In the Figure (7) we can see the evolution during the simulation time of the number of particles in each node.

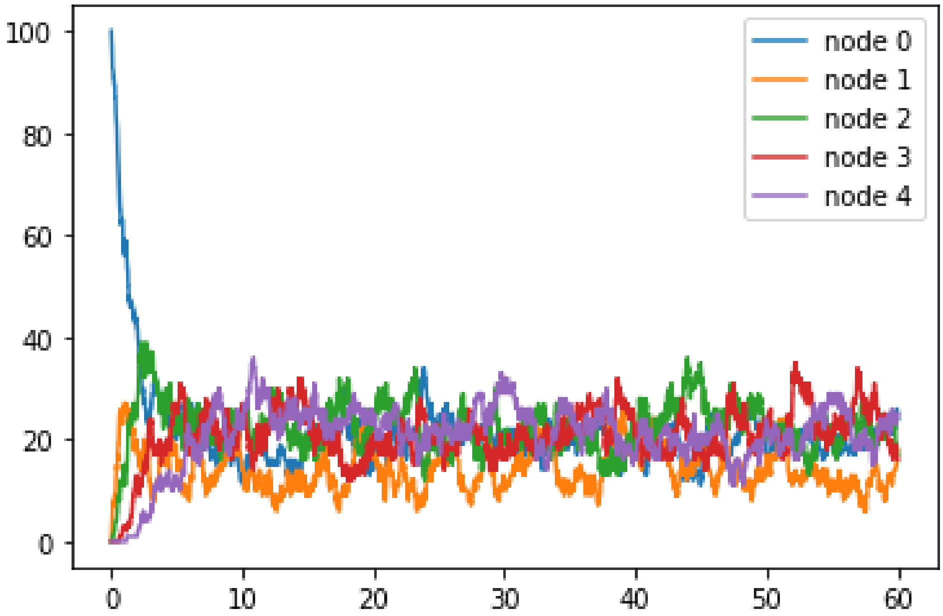


Figure 7: Number of particles in node during time of simulation

The average number of particle in each node at the end of the simulations is

$$\begin{array}{ccccc} o & a & b & c & d \\ \end{array}$$
$$\begin{bmatrix} 18.74 & 14.92 & 22.50 & 22.30 & 21.54 \end{bmatrix}$$

If we divide the obtained average by the number of particles put in the graph we obtain

$$\begin{array}{ccccc} o & a & b & c & d \\ \end{array}$$
$$\begin{bmatrix} 0.1874 & 0.1492 & 0.2250 & 0.2230 & 0.2154 \end{bmatrix}$$

that is pretty similar to the stationary distribution of the continuous-time random walk followed by the single particles computed in Problem 1.

$$\begin{bmatrix} 0.1851 & 0.1481 & 0.2222 & 0.2222 & 0.2222 \end{bmatrix}$$

Problem 3

We consider now the open network in Figure (2) with transition rate matrix Λ_{open} in (2). Particles will enter the system at node o according to a Poisson process with rate $\lambda = 1$ and each node will then pass along a particle according to a specific rate.

a) **Proportional rate:**
In this first part the rate of the Poisson clock of each node is equal to the number of particles in it. The system has been simulated for 60 time units giving as result the evolution in Figure (8a), then the input rate λ has been increased at 10, 100 and 500 obtaining the evolution in Figures (8b) (8c) (8d). The system does not blow up, in fact we can see how particles keep distributing across all nodes even when the input rate λ increases.

b) **Proportional rate:**
In this part the rate of the Poisson clock of each node is fixed, and equal to one. Again we simulate it for 60 time units obtaining the distribution of particles in Figure (9a). The the input rate has been increased to 10 and 100 with result in Figures (9b) (9c). We can notice that with input rate 100 the particles are all accumulated in the starting node o and have difficulty to distribute across other nodes, in particular we have 0 particles in node d. The system has blown up now.

After a further investigation we can find that this behavior of the system starts with an input rate $\lambda = 30$ when about the 80% of particles in the system at the end of the simulation is still in the node o, as we can see in Figure (10).

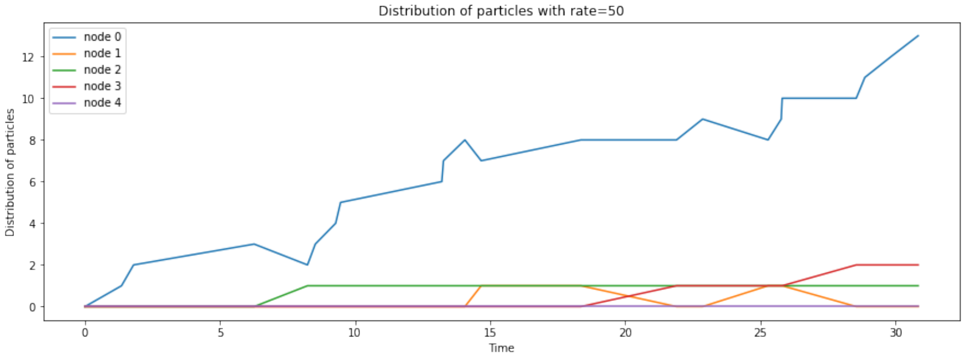
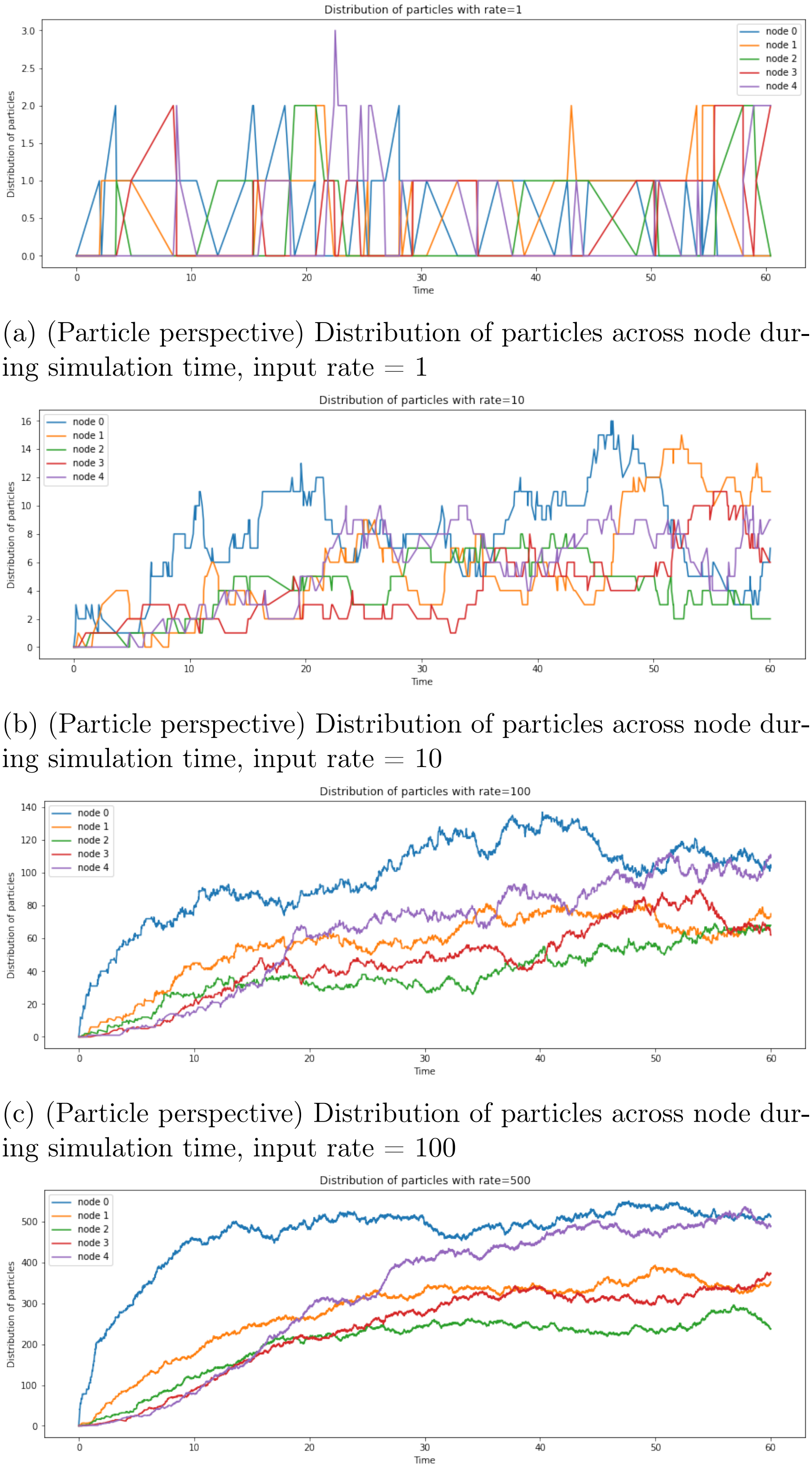


Figure 10: (Node perspective) System starts blowing up with input rate $\lambda = 30$



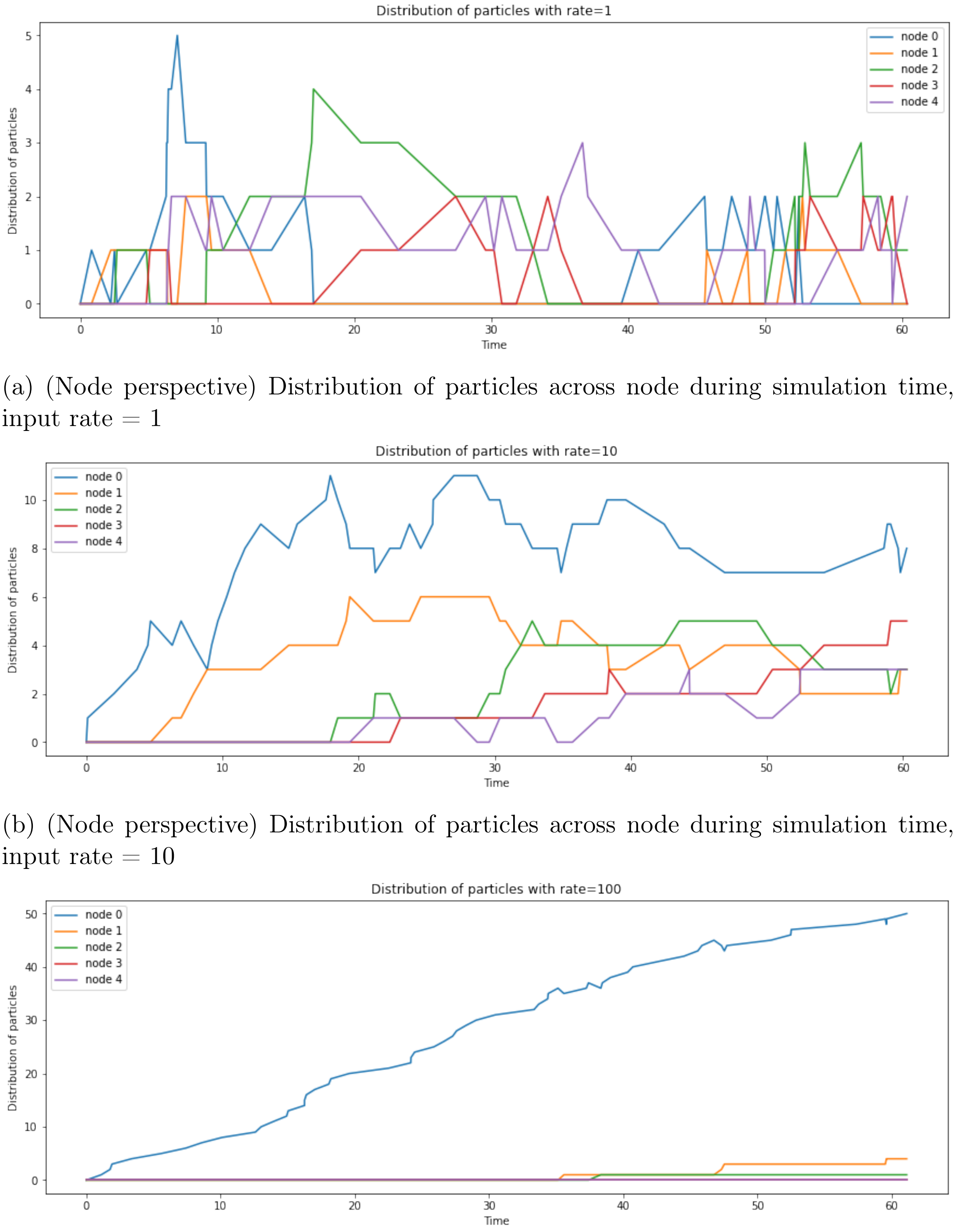
(a) (Particle perspective) Distribution of particles across node during simulation time, input rate = 1

(b) (Particle perspective) Distribution of particles across node during simulation time, input rate = 10

(c) (Particle perspective) Distribution of particles across node during simulation time, input rate = 100

(d) (Particle perspective) Distribution of particles across node during simulation time, input rate = 500

Figure 8



(a) (Node perspective) Distribution of particles across node during simulation time, input rate = 1

(b) (Node perspective) Distribution of particles across node during simulation time, input rate = 10

(c) (Node perspective) Distribution of particles across node during simulation time, input rate = 100

Figure 9