Homework 2 - Network Dynamics and Learning

Tommaso Mazzarini s334004
Network Dynamics and Learning
December 2024

Problem 1: Continuous-Time Random Walk and Opinion Dynamics

The first problem of this homework is divided into two parts. The initial part focuses on analyzing the behavior of a single particle performing a continuous-time random walk in a network. The network is represented by the graph G, as shown in Figure 1, with nodes labeled o, a, b, c, and d. The particle's movement between nodes is governed by the transition rate matrix Λ, as described below:

$$\Lambda = \begin{array}{cc} \begin{bmatrix} 0 & \frac{2}{5} & \frac{1}{5} & 0 & 0 \\ 0 & 0 & \frac{3}{4} & \frac{1}{4} & 0 \\ \frac{1}{2} & 0 & 0 & \frac{1}{3} & 0 \\ 0 & 0 & \frac{1}{3} & 0 & \frac{2}{3} \\ 0 & \frac{1}{2} & 0 & \frac{1}{3} & 0 \end{bmatrix} & \begin{array}{c} o \\ a \\ b \\ c \\ d \end{array} \end{array} \tag{1}$$

The corresponding graph is depicted in Figure 1.

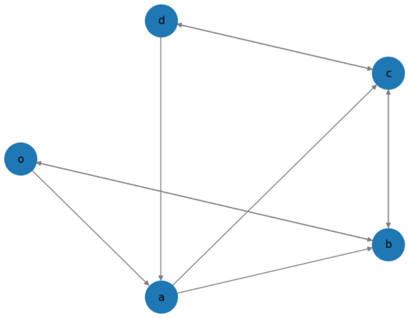

Figure 1: Closed network in which a particle moves according to the transition rate matrix Λ.

The goal of the first part of this problem is to simulate the continuous-time random walk and analyze key properties of the particle's behavior within the network. Specifically, the following tasks are addressed:

(a) Compute the average time it takes for a particle starting at node a to leave the node and then return to it, using simulations.

(b) Compare the simulated return time to the theoretical return time $E_a[T_a^+]$.

(c) Determine the average time it takes for a particle to move from node o to node d using simulations.

(d) Compare the simulated hitting time to the theoretical hitting time $E_o[T_d]$.

To help with the simulations, it is useful to recall the behavior of Poisson processes. Specifically, for a rate-r Poisson process, the inter-arrival time t_{next} between two consecutive events follows an exponential distribution with rate r. If $u \sim \mathcal{U}(0,1)$ is a uniformly distributed random variable, the time t_{next} can be computed as:

$$t_{\text{next}} = -\frac{\ln(u)}{r}. \tag{2}$$

Additionally, the memoryless property of the exponential distribution states that:

$$P(X \geq t+s \mid X \geq t) = \frac{P(X \geq t+s)}{P(X \geq t)} = \frac{e^{-r(t+s)}}{e^{-rt}} = e^{-rs} = P(X \geq s). \tag{3}$$

The second part of the problem involves opinion dynamics on the same graph. In particular, we interpret the matrix Λ as the weight matrix of a graph $G = (V, E, \Lambda)$ and analyze the French-DeGroot dynamics. The tasks include:

(e) Simulate the French-DeGroot dynamics with an arbitrary initial condition $x(0)$ and determine whether the dynamics converge to a consensus state.

(f) Compute the variance of the consensus value, given that the initial state of the dynamics for each node $i \in V$ is $x_i(0) = \xi_i$, where $\{\xi_i\}_{i \in V}$ are independent random variables. The variances of these random variables are specified as $\sigma_a^2 = \sigma_b^2 = 2$ and $\sigma_c^2 = \sigma_d^2 = 1$. Finally, compare the computed variance with the results from numerical simulations.

(g) Analyze the asymptotic behavior of the dynamics after removing the edges (d,a), (d,c), (a,c), and (b,c) from the graph. Describe the resulting behavior of the system as it evolves over time. If the dynamics converge to an asymptotic state, explain how this final state relates to the initial condition $x(0)$ and provide motivation for your observations.

(h) Consider the modified graph (V, E, Λ), where the edges (b,o) and (d,a) are removed. Study the French-DeGroot dynamics on this new graph and analyze how the asymptotic behavior of the system changes based on the initial condition $x(0)$. Provide a detailed motivation for how the asymptotic state evolves in terms of the initial condition.

a) Simulated Return Time for Node a

To compute the expected time for a particle starting from node a to leave and then return to node a under the given transition rate matrix Λ, there are three possible approaches:

1. **Global Poisson Clock:**
   A global Poisson clock ticks with a rate $\omega^* = \max_i(\omega_i)$, where $\omega_i = \sum_j \Lambda_{ij}$

is the total rate of leaving node i. At each tick, the particle at node i transitions to a neighboring node $j \neq i$ with probability:

$$\bar{P}_{ij} = \frac{\Lambda_{ij}}{\omega^*}, \quad i \neq j, \tag{4}$$

and remains at i with probability:

$$\bar{P}_{ii} = 1 - \sum_{j \neq i} \bar{P}_{ij}. \tag{5}$$

2. **Independent Node Clocks:**
   Each node i is equipped with an independent Poisson clock ticking at rate $\omega_i = \sum_j \Lambda_{ij}$. When the clock at node i ticks, the particle transitions to node $j \neq i$ with probability:

$$P_{ij} = \frac{\Lambda_{ij}}{\omega_i}. \tag{6}$$

3. **Independent Link Clocks:**
   Each link (i, j) has an independent Poisson clock ticking at rate $\omega_{(i,j)} = \Lambda_{ij}$. If the clock of link (i, j) ticks while the Markov chain is at node i, the particle moves to node j.

Although all three approaches yield the same results, we adopt the **global Poisson clock** method for its simplicity in implementation. To simulate the particle's movement, we derive the transition probability matrix $\bar{P}$ from the transition rate matrix Λ using the equations above: (4) and (5).

The **inter-arrival time** t_{next} between ticks of the global clock is sampled from an exponential distribution with rate ω^*. Specifically, we generate a uniformly distributed random variable $u \sim U(0, 1)$ and compute:

$$t_{\text{next}} = -\frac{\ln(u)}{\omega^*}. \tag{7}$$

At each tick, the particle transitions from its current node i to a neighboring node j based on the transition probabilities $\bar{P}_{ij}$. This process is repeated until the particle returns to the starting node a for the first time.

To estimate the expected return time $E_a[T_a^+]$, we perform $N = 10,000$ independent simulation trials. In each trial k, the total return time T_k is recorded. The average return time is then computed as:

$$E_a[T_a^+] = \frac{1}{N} \sum_{k=1}^{N} T_k. \tag{8}$$

The result of the simulation yields an approximate expected return time: $E_a[T_a^+] \approx 6.036$.

b) Comparison Between Simulated and Theoretical Return Times

In problem (a), we estimated the expected return time $E_a[T_a^+]$ for a particle starting at node a using simulations. In this section, we aim to compute $E_a[T_a^+]$

from a theoretical perspective. To do so, we first determine the invariant probability vector $\bar{\pi}$ of the continuous-time Markov chain (CTMC) associated with the given transition rate matrix Λ.

The probability distribution $\bar{\pi}(t)$ of the CTMC $X(t)$ with transition rate matrix Λ is defined as

$$\bar{\pi}_i(t) = P(X(t) = i), \quad i \in X. \tag{9}$$

This distribution evolves according to the following differential equation:

$$\frac{d}{dt}\bar{\pi}(t) = -L'\bar{\pi}(t), \tag{10}$$

where

$$L = \text{diag}(w) - \Lambda, \quad w = \Lambda\mathbf{1}.$$

The invariant probability vector $\bar{\pi}$ is the eigenvector of L' corresponding to the eigenvalue 0. Alternatively, $\bar{\pi}$ can be obtained as the dominant left eigenvector of the row-stochastic matrix $\bar{P}$.

By solving the eigenvector problem for $\bar{P}$, we compute its dominant eigenvector corresponding to the eigenvalue 1. The result is:

$$\bar{\pi} = [0.23058252, 0.16504854, 0.27669903, 0.18203883, 0.14563107].$$

We now proceed to compute the theoretical return time. For a continuous-time Markov chain, the expected return time to node i can be determined using the formula:

$$E_i[T_i^+] = \frac{1}{\bar{\omega}_i\bar{\pi}_i}. \tag{11}$$

Applying this formula for node a, the return time is calculated as:

$$E_a[T_a^+] = \frac{1}{\bar{\omega}_a\bar{\pi}_a} \approx 6.058.$$

This theoretical result aligns closely with the value obtained in problem (a) via simulation, where we estimated $E_a[T_a^+] \approx 6.036$. The minor discrepancy between the two values can be attributed to the randomness inherent in the simulation process and the finite number of trials conducted.

As illustrated in Figure 2, increasing the number of simulations results in an asymptotic convergence of the experimental values toward the theoretical value, in accordance with the law of large numbers.

c) Simulated Hitting Time from Node o to Node d

In problem (c), we investigate the hitting time $E_o[T_d]$, which represents the expected time for a continuous-time Markov process to transition from a starting node o to a target node d. The experimental procedure for this part follows a method similar to that in problem (a), with the key distinction being that in this case, the particle starts at node o instead of node a.

During the simulation, we track the time it takes for the particle to reach node d for the first time, rather than returning to the starting node o. Apart from this change, the setup is identical to the procedure in problem (a).

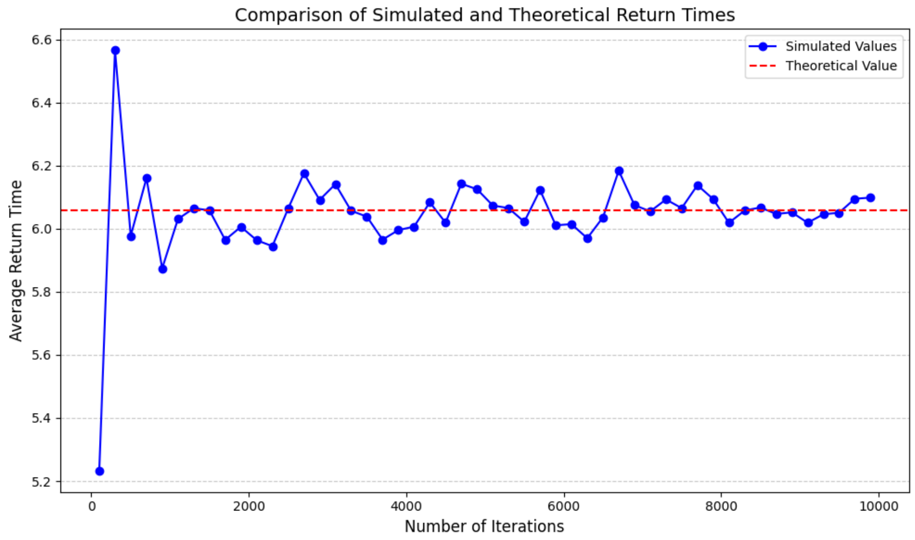

Figure 2: Evolution of the average return-time over time compared to the theoretical value.

In this regard, we simulate the particle's movement through the network using the global Poisson clock method, which relies on the transition rate matrix Λ and the associated transition probabilities.

The inter-arrival time t_{next} between ticks of the global Poisson clock is sampled from an exponential distribution with rate $\omega^* = \max_i(\omega_i)$, where $\omega_i = \sum_j \Lambda_{ij}$ represents the total rate of departure from node i. As in the previous step, we generate a random variable $u \sim U(0,1)$ and compute the time between events using equation (7).

At each tick of the clock, the particle moves from its current node i to a neighboring node j with transition probabilities $\bar{P}_{ij} = \frac{\Lambda_{ij}}{\omega^*}$. This process continues until the particle first reaches node d.

To estimate the expected hitting time $E_o[T_d]$, we run $N = 10,000$ independent simulation trials. In each trial k, the total time T_k it takes for the particle to move from node o to node d is recorded. The average hitting time is then calculated as:

$$E_o[T_d] = \frac{1}{N}\sum_{k=1}^{N} T_k.$$

The simulation results indicate that the average time for the particle to travel from node o to node d is approximately 10.859.

d) Comparison Between Simulated and Theoretical Hitting Times

In problem (d), we focus on calculating the theoretical hitting time $E_o[T_d]$, which represents the expected time for a particle starting at node o to reach node d.

For a continuous-time Markov chain, the expected hitting time $E_i[T_s]$ from node i to the target set $S = \{d\}$ satisfies the following equation:

$$E_i[T_s] = \begin{cases} 0, & \text{for } i \in S. \\ \frac{1}{w_i} + \sum_{j \notin S} P_{ij} E_j[T_s], & \text{for } i \notin S, \end{cases} \tag{12}$$

where w_i is the total exit rate from node i, P_{ij} is the transition probability from node i to node j, and $S = \{d\}$ is the target set of nodes.

The total exit rate w_i is given by $w_i = \sum_j \Lambda_{ij}$, where Λ_{ij} are the elements of the transition rate matrix Λ.

Using this formula, we computed the theoretical hitting time $E_o[T_d]$, which resulted in a value of approximately 10.766. This theoretical result is in good agreement with the simulation results from problem (c), where the average time for a particle to travel from node o to node d was found to be 10.858. The small difference between these two values, also in this case, can be attributed to the inherent randomness and variability in the simulation process.

To further validate this, we plotted the simulated hitting times as a function of the number of simulations. As shown in Figure 3, the simulated hitting time converges to the theoretical value as the number of simulations increases. This demonstrates the asymptotic behavior of the simulated hitting time towards the theoretical hitting time.

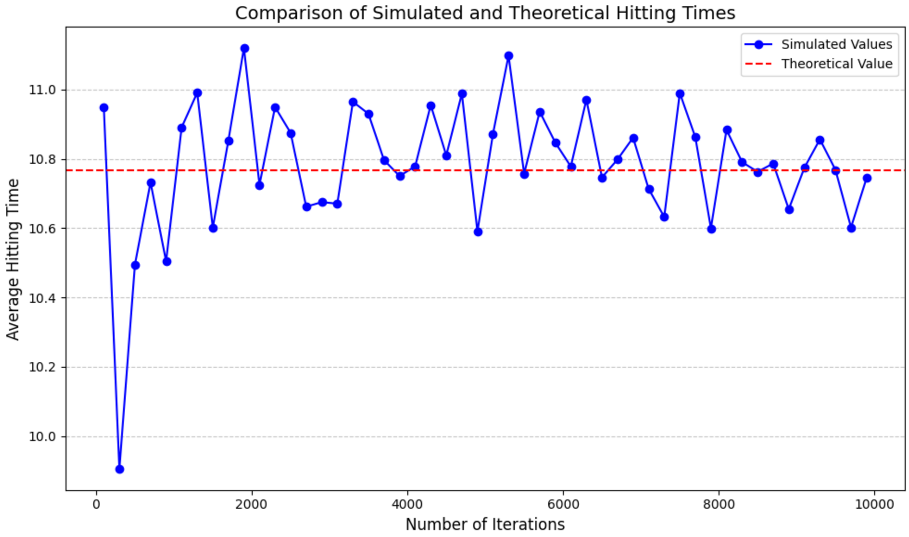

Figure 3: Evolution of the average hitting-time over time compared to the theoretical value.

e) Simulation of French-DeGroot Dynamics and Consensus Behavior

In this exercise, we simulate the French-DeGroot dynamics on a graph $G = (V, E, \Lambda)$, as shown in Figure 4, where Λ is the weight matrix, also called trust

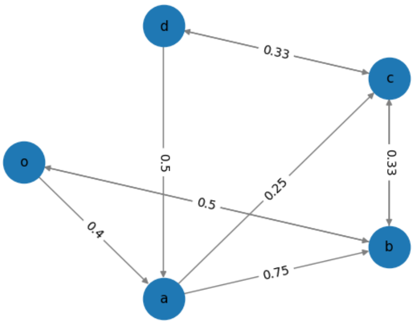

Figure 4: Graph representation of the network used for simulating the French-DeGroot dynamics.

matrix. The goal is to determine whether the dynamics always converge to a consensus state, regardless of the initial condition $x(0)$.

The French-DeGroot dynamics are governed by the equation:

$$x(t+1) = Px(t), \tag{13}$$

where $x(t) \in \mathbb{R}^V$ represents the opinions of the nodes at time t, and P is the normalized adjacency matrix. The matrix P is computed as:

$$P = D^{-1}A, \tag{14}$$

where A is the adjacency matrix (corresponding to the weight matrix Λ), and D is the degree matrix, which contains the sum of outgoing edge weights for each node. In this model, each node updates its opinion based on the opinions of its neighbors, meaning that the opinion of node i is influenced by the opinions of nodes to which it is connected by directed edges.

For the system to converge to a consensus, it must satisfy the conditions of strong connectivity, and the presence of a single globally reachable and aperiodic strongly connected component. If these conditions are met, the dynamics will converge to a consensus. More specifically, the system converges to:

$$\lim_{t\to\infty} x(t) = \alpha\mathbf{1}, \tag{15}$$

where α is a scalar value representing the consensus opinion. This value can be determined as a weighted average of the initial opinions, with weights given by the components of the normalized left dominant eigenvector π of the transition matrix P, which satisfies $P'\pi = \pi$. The consensus value α is then given by:

$$\alpha = \pi' x(0), \tag{16}$$

where $x(0)$ is the initial opinion vector.

To simulate the French-DeGroot dynamics, we begin by constructing the graph G from the matrix Λ. An edge is added between two nodes if the corresponding weight in Λ is greater than zero. We then compute the transition matrix P, which governs the evolution of opinions over time.

Before running the simulation, we verify that the graph is strongly connected and that its strongly connected component, which is globally reachable, is aperiodic. Strong connectivity ensures that there is a direct path between every pair of nodes, which is crucial for consensus since it allows the influence to propagate throughout the network. Additionally, the condensation graph must have a single sink component, guaranteeing that the system will eventually converge to a unified opinion. The aperiodicity of the globally reachable strongly connected component ensures that there are no regular ”cycles” in the graph that would obstruct convergence.

We then run 100 trials with randomly chosen initial opinions. In each trial, we iteratively apply the transition matrix P to the initial opinion vector and check whether the opinions converge to the same value. If all 100 trials lead to convergence to the same opinion, we conclude that the dynamics always lead to consensus, regardless of the initial condition.

The results of our simulation confirm that the dynamics always converge to a consensus state, regardless of the initial condition $x(0)$. This demonstrates that the graph satisfies the necessary conditions for consensus. Figure 5 shows the convergence of opinions in one simulation, where the opinions evolve over time and asymptotically converge to a single value.

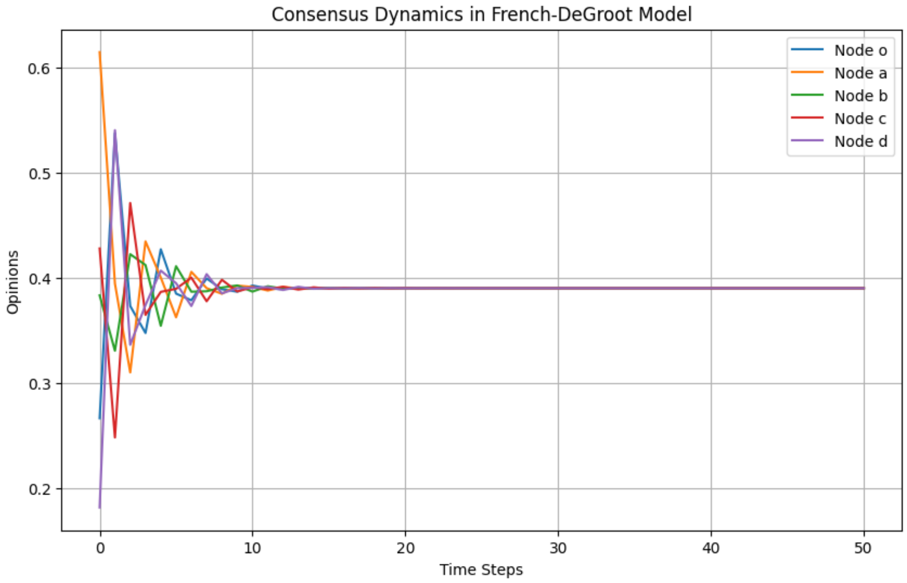

Figure 5: Opinion convergence in the French-DeGroot dynamics simulation.

f) Variance of the Consensus Value and Numerical Comparison

In this section, we consider the French-DeGroot dynamics on a strongly connected and aperiodic graph $G = (V, E, \Lambda)$, where each node's initial state $x_i(0)$ is an independent random variable. Specifically, we assume the initial variances of the nodes are given by:

$$\sigma_a^2 = \sigma_b^2 = \sigma_c^2 = 2, \sigma_o^2 = \sigma_d^2 = 1. \tag{17}$$

The goal is to compute the variance of the consensus value x^* and compare the theoretical result with numerical simulations.

$$x^* = \sum_{i \in V} \pi_i x_i(0), \tag{18}$$

where π_i is the entry of the stationary distribution for node i, and $x_i(0)$ is the initial state of node i.

Since the initial states $x_i(0)$ are independent random variables, the variance of the consensus value x^* can be computed as the sum of the variances of the initial states, weighted by the square of the corresponding entries in the stationary distribution π:

$$\mathrm{Var}(x^*) = \mathrm{Var}\left(\sum_{i \in V} \pi_i x_i(0)\right) = \sum_{i \in V} \pi_i^2 \mathrm{Var}(x_i(0)). \tag{19}$$

This follows from the linearity of variance for independent random variables and the property that $\mathrm{Var}(aX) = a^2 \mathrm{Var}(X)$ for a constant a. Substituting the given initial variances:

$$\mathrm{Var}(x^*) = \pi_o^2 \sigma_o^2 + \pi_a^2 \sigma_a^2 + \pi_b^2 \sigma_b^2 + \pi_c^2 \sigma_c^2 + \pi_d^2 \sigma_d^2. \tag{20}$$

Using the initial variances $\sigma_o^2 = \sigma_d^2 = 1$ and $\sigma_a^2 = \sigma_b^2 = \sigma_c^2 = 2$, we obtain:

$$\mathrm{Var}(x^*) = \pi_o^2(1) + \pi_a^2(2) + \pi_b^2(2) + \pi_c^2(2) + \pi_d^2(1). \tag{21}$$

Thus, the theoretical variance of the consensus value is computed to be: $\mathrm{Var}(x^*) = 0.3717$.

To verify this theoretical result, we perform numerical simulations. We simulate 10,000 trials of the French-DeGroot dynamics, each with randomly chosen initial opinions. In each trial, we run the dynamics for 50 steps, updating the opinions using the transition matrix P. After each simulation, we compute the consensus value and track its variance over the 10,000 steps.

The simulated variance of the consensus value obtained from these trials is: $\mathrm{Var}_{\mathrm{sim}}(x^*) = 0.3737$.

By comparing the theoretical variance 0.3717 with the simulated variance 0.3737, we observe that the two values are in close agreement, indicating that the theoretical calculation accurately predicts the behavior of the system. The small difference between the theoretical and simulated variances can be attributed to the stochastic nature of the simulations and the finite number of trials.

g) Asymptotic Behavior After Edge Removal in French-DeGroot Dynamics

In this exercise, we remove four edges from the original network described in point f: (d, a), (d, c), (a, c), and (b, c), leading to a significant change in the network's structure and dynamics, as shown in Figure 6. The goal is to examine the asymptotic behavior of the system after these modifications, specifically

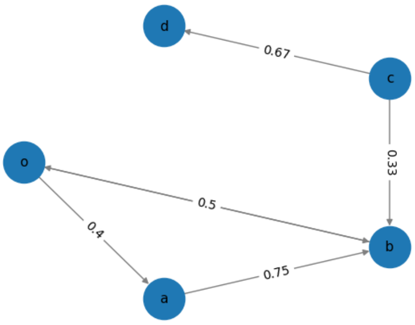

Figure 6: Network structure after removing edges (d, a), (d, c), (a, c), and (b, c).

investigating whether consensus can be achieved and how the initial conditions influence the final state of the system.

After removing the specified edges, the graph is no longer strongly connected. Therefore, we analyze the condensation graph of the network, which reveals two distinct sink components, indicating that global consensus cannot be achieved. The first component, $\{o, a, b\}$, forms a strongly connected component (SCC) and is aperiodic, meaning it is capable of reaching a local consensus. The second component consists of the isolated node d, which remains unaffected by other nodes. The condensation graph highlights that node d maintains its initial opinion, preventing the system from achieving global consensus.

The dynamics within the component $\{o, a, b\}$ are driven by the consensus value determined by the initial states of the nodes and the stationary distribution of the corresponding random walk. Specifically, the consensus value for this component is given by:

$$x_{\text{consensus}} = \mathbf{x}_0[0:3] \cdot \pi[0:3], \tag{22}$$

where $\mathbf{x}_0[0:3]$ represents the initial opinions of nodes o, a, and b, and $\pi[0:3]$ is the stationary distribution of the SCC. The nodes in this component converge to a consensus value based on their initial opinions and the relative positions within the network.

Node d, having zero out-degree, does not interact with any other node and thus remains isolated within the network, preserving its initial opinion indefinitely. As a result, the opinion of node d at time $t \geq 1$ remains unchanged:

$$x_d = x_d(0) \quad \text{for} \quad t \geq 1. \tag{23}$$

Node c is influenced by both the strongly connected component $\{o, a, b\}$ and node d. To explore the behavior of node c, we consider the case where node d is initialized with an opinion of 0. In this case, node d does not influence node c, and node c is solely influenced by the consensus of the component $\{o, a, b\}$. Analyzing the transition probability matrix P, we find that node c's opinion will eventually converge to one-third of the consensus value of the SCC $\{o, a, b\}$, since node c is primarily influenced by node b. The final state of node c is given by:

$$x_c \to \frac{x_{\text{consensus}}}{3} = \frac{\mathbf{x}_0[0:3] \cdot \pi[0:3]}{3}. \tag{24}$$

To validate these theoretical predictions, we conducted numerical simulations. In these simulations, 100 random initial conditions were tested, and the French-DeGroot dynamics were run for 100 steps per trial. The results were consistent with the theoretical analysis: nodes o, a, and b reached consensus, while node d retained its initial opinion, as shown in Figure 7. Moreover, when node d was initialized with an opinion of 0, the final opinion of node c was approximately one-third of the consensus value of the component $\{o, a, b\}$, confirming the theoretical predictions.

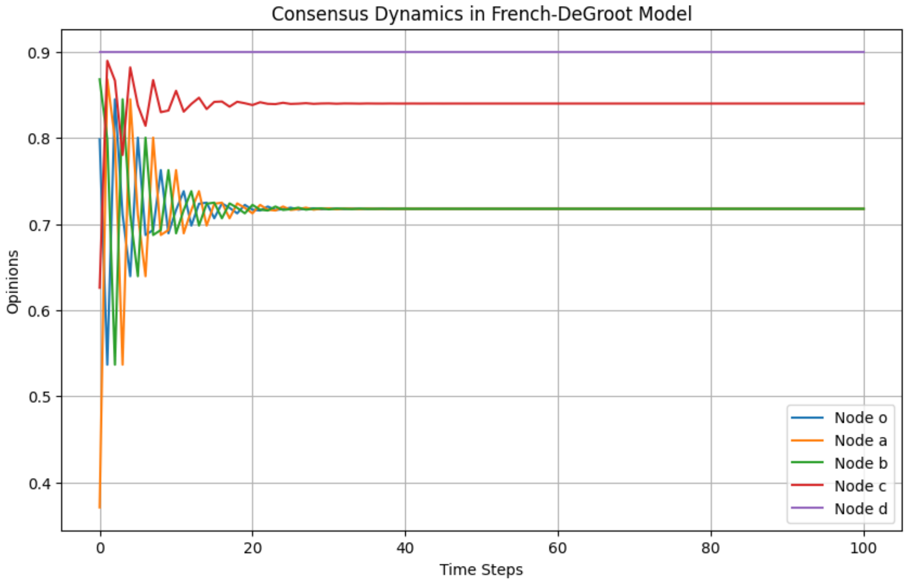

Figure 7: Simulation results showing the consensus behavior of the network.

h) Modified Graph Analysis and Evolution of Asymptotic States

In this exercise, we analyze the French-DeGroot dynamics on a modified graph $G = (V, E, \Lambda)$, which is obtained by removing the edges (b, o) and (d, a), as illustrated in Figure 8. The removal of these edges results in the graph losing its strong connectivity. To better understand the dynamics of the system, we construct the condensation graph, which reveals three strongly connected components (SCCs):

- {b, c, d}, a “sink” component with no outgoing edges;
- {a}, which points to SCC0;
- {o}, which points to both SCC0 and SCC1.

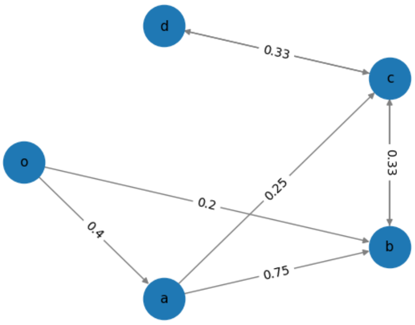

Figure 8: Modified network graph after removing edges (b, o) and (d, a).

The structure of these SCCs implies that the sink component $\{b, c, d\}$ exerts a dominant influence on the long-term behavior of the system, as all other components are influenced by it either directly or indirectly.

However, this sink component exhibits periodic behavior due to the symmetric interactions among nodes b, c, and d, preventing local consensus within this component. Specifically, after the first iteration, nodes b and d adopt the opinion of node c, while node c updates its opinion as the average of nodes b and d. This leads to an oscillatory pattern with a periodicity of 2.

Node a is influenced by the opinions of nodes b and c from the sink component. The update rule for node a is given by:

$$\text{opinion}[a, t] = \frac{3}{4} \cdot \text{opinion}[b, t-1] + \frac{1}{4} \cdot \text{opinion}[c, t-1]. \tag{25}$$

Given the periodic nature of the opinions of nodes b and c, node a's opinion will also oscillate with the same periodicity, meaning that node a cannot achieve a steady consensus. Similarly, node o is influenced by both a and b, and its opinion inherits the oscillatory dynamics of these nodes, resulting in the same periodic behavior:

$$\text{opinion}[o, t] = \frac{2}{3} \cdot \text{opinion}[a, t-1] + \frac{1}{3} \cdot \text{opinion}[b, t-1]. \tag{26}$$

The oscillatory behavior of the system is illustrated in the following graphs (Figure 9 and Figure 10), which depict the French-DeGroot dynamics of the various nodes.

The system's ability to reach consensus depends critically on the initial opinions of the nodes. When the initial opinions of nodes b, c, and d are identical, the oscillations within the sink component disappear, allowing the system to converge to a stable consensus. In this case, the consensus value propagates through the rest of the graph, ultimately leading to a global consensus state. However, if the initial opinions of nodes b, c, and d differ, the oscillations persist, preventing the system from reaching global consensus.

This behavior is confirmed through numerical experiments conducted with the following initial opinions:

$$x(0) = [0.2, 1, 0.5, 0.5, 0.5] \tag{27}$$

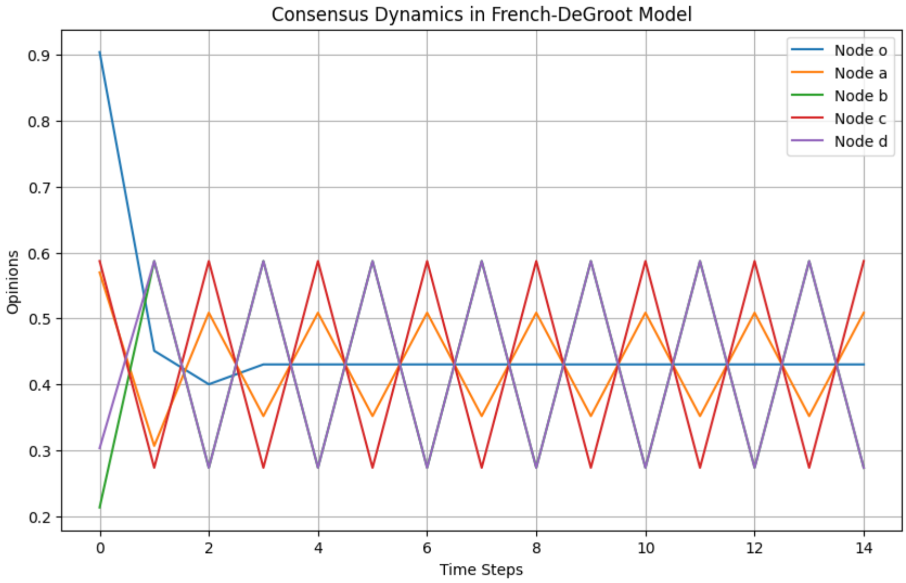

Figure 9: Zoomed-in view of the oscillatory behavior of the French-DeGroot dynamics in the modified network.

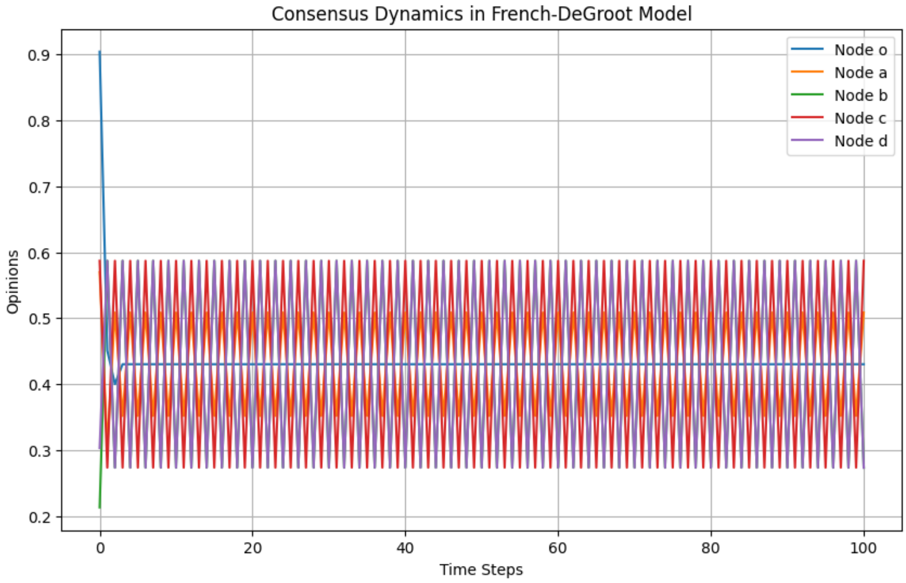

Figure 10: Oscillatory behavior of the French-DeGroot dynamics in the modified network.

In both cases, when the initial opinions of nodes b, c, and d were identical, the system successfully converged to a stable consensus. Figure 11 illustrates this stable consensus, which occurs when the opinions of nodes b, c, and d are aligned.

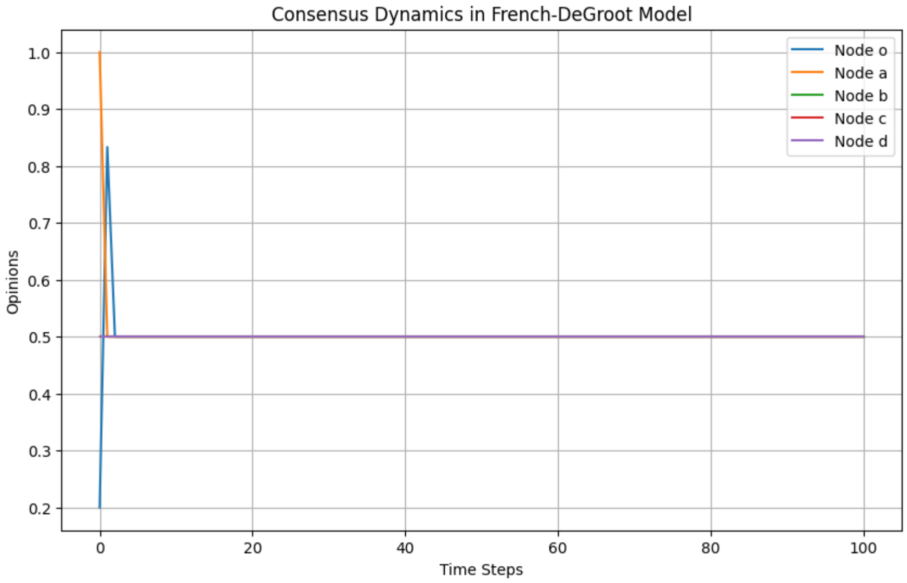

Figure 11: Stable consensus achieved when nodes b, c, and d have identical initial opinions.

Problem 2: Multi-Particle Dynamics in a Continuous-Time Random Walk

The second problem of this homework focuses on analyzing the behavior of multiple particles performing a continuous-time random walk in the network depicted in Figure **??**. The network is described by the graph $G = (V, E, \Lambda)$, where the edge weights are given by the transition rate matrix Λ. Each particle in the network moves according to the following stochastic rules:

- The time a particle spends at node i is exponentially distributed with a mean of $1/\omega_i$, where $\omega_i = \sum_j \Lambda_{ij}$ is the total rate of outgoing transitions from node i.
- Upon departure from node i, the particle transitions to a neighboring node j according to the normalized transition probability matrix $P = \text{diag}(\omega)^{-1}\Lambda$, where ω is the vector of outgoing rates from all nodes.

Instead of considering a single particle, as in Problem 1, we now study the system with $N = 100$ particles moving in the network in continuous time. The goal is to analyze the system from two different perspectives: the **particle perspective** and the **node perspective**. Specifically, the following tasks are addressed:

(a) **Particle perspective:**

- Simulate the movement of $N = 100$ particles, all starting at node a.
- Compare this result with the return time obtained in Problem 1 for a single particle.

- Analyze and explain any similarities or differences observed.

(b) **Node perspective:**

- Simulate the system with $N = 100$ particles, all starting at node a, over a total simulation time of 60 time units.
- Illustrate the evolution of particle distributions across nodes using a plot showing the number of particles in each node over time.
- Compare the simulation results with the stationary distribution of the continuous-time random walk for a single particle.
- Discuss any agreements or deviations observed.

a) Particle perspective

To analyze the system from the particle perspective, we simulate the movement of $N = 100$ particles, all initially positioned at node a. This simulation mirrors the setup from Problem 1, where we tracked the return time of a single particle to its starting node. However, in this case, at each time step, we randomly select one of the N particles to move. The selected particle transitions to a neighboring node according to the transition probability matrix $\bar{P}$, with the time spent at each node following an exponential distribution.

This process continues until all N particles return to node a, at which point we calculate the average return time for all particles. To ensure a robust estimate, we repeat this procedure $10,000$ times.

The results demonstrate that the average return time for a particle to return to node a remains consistent with the result obtained in Problem 1 for a single particle:

$$\mathbb{E}_a^{(N)}[T_a] = \mathbb{E}_a[T_a] \approx 6.06. \tag{28}$$

This agreement can be attributed to the apparent independence of the particles, as suggested by the results. Despite simulating N particles simultaneously, their movements behave as if they are independent of each other. Each particle follows its own path without being influenced by the others, which results in the system effectively behaving like N interleaved single-particle systems. As a result, the scaling factor $\frac{1}{N}$ in the transition rates only affects the time scaling of the process, without altering the underlying dynamics. This confirms that the return probability distribution remains unchanged, reflecting the independent behavior of the particles.

b) Node perspective

To analyze the system from the node perspective, we simulate the behavior of $N = 100$ particles, all starting at node a, over a time horizon of 60 time units. The simulation uses a system-wide Poisson clock with rate $N\omega^*$, where ω^* is the maximum sum of outgoing edge weights obtained in Exercise 1.

At each tick of the Poisson clock:

- A node is selected randomly, with the probability of selection proportional to the number of particles currently at that node.

- A particle is moved from the selected node to a neighboring node based on the transition probability matrix $\bar{P}$.
- The particle count of the departure node is decreased by 1, while the particle count of the arrival node is increased by 1.

Throughout the simulation, we keep track of the number of particles at each node over time. At the end of the simulation, we obtain the final distribution of particles across the nodes.

After simulating for 60 time units, the final number of particles at nodes o, a, b, c, d are approximately: $[27, 14, 24, 19, 16]$.

The expected stationary distribution for the single-particle case is given by:

$$\bar{\pi} = [0.2306, 0.1650, 0.2767, 0.1820, 0.1456]. \tag{29}$$

Multiplying this distribution by $N = 100$, the expected stationary particle counts are:

$$N\bar{\pi} = [23.06, 16.50, 27.67, 18.20, 14.56]. \tag{30}$$

The simulation results show that the final particle distribution closely matches the expected stationary distribution $N\bar{\pi}$. Small deviations are expected due to the finite simulation time and the stochastic nature of the process, but the overall agreement demonstrates that the particles' behavior aligns with the asymptotic stationary distribution predicted by the continuous-time random walk.

The Figure 12 illustrates the evolution of the number of particles in each node during the simulation time. As the simulation progresses, the system converges toward the stationary distribution $\bar{\pi}$.

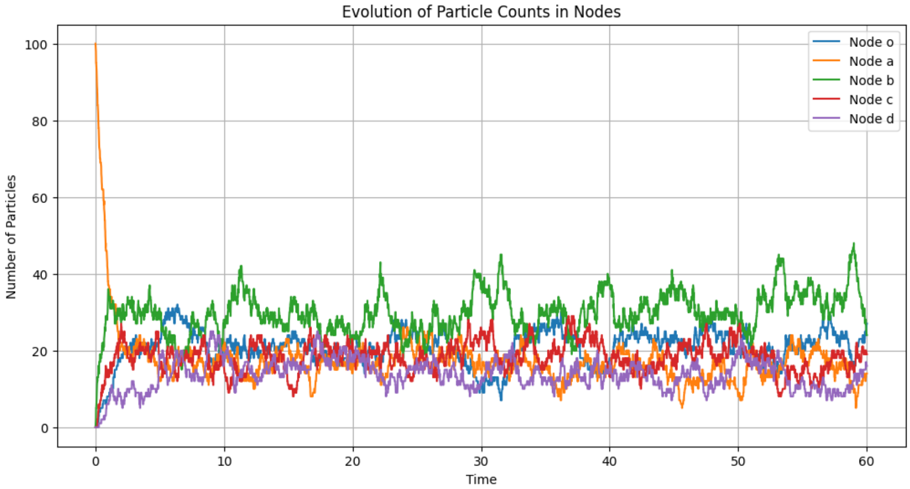

Figure 12: Evolution of the number of particles in each node over 60 time units.

This analysis confirms that the asymptotic behavior of the multi-particle system matches the stationary distribution of the single-particle random walk. By simulating the evolution of the particle counts over time, we observe convergence toward the expected equilibrium distribution $\bar{\pi}$, scaled by the number of particles N.

Problem 3: Simulation of Particle Dynamics in an Open Network

In this problem, we analyze the behavior of particles in an open network characterized by the transition rate matrix Λ_{open}, given in equation (31) and represented in Figure 13. The network comprises five nodes: o, a, b, c, and d, where particles enter through the input node o at a rate λ, modeled as a Poisson process.

$$\Lambda_{\text{open}} = \left[\begin{array}{ccccc} 0 & \frac{3}{4} & \frac{3}{4} & 0 & 0 \\ 0 & 0 & \frac{1}{4} & \frac{1}{4} & \frac{2}{4} \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{array}\right] \quad \begin{array}{c} o \\ a \\ b \\ c \\ d \end{array} \tag{31}$$

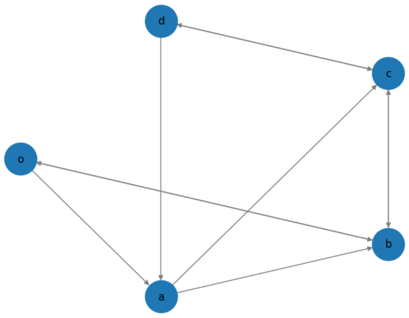

Figure 13: Representation of the open network associated with Λ_{open}.

The objectives of the analysis are as follows:

1. **Proportional rate scenario:** Simulate the system for 60 time units with $\lambda = 100$. Track the evolution of the particle count in each node and determine the maximum input rate λ that the system can sustain without becoming unstable.

2. **Fixed rate scenario:** Simulate the system for 60 time units starting with $\lambda = 1$. Gradually increase λ to identify the critical threshold at which the particle count grows unbounded, and explain the resulting system behavior.

a) Proportional Rate

The objective, of this first task, is to simulate the system under the proportional rate scenario for 60 time units, analyze the evolution of the particle distribution over time, and determine the largest input rate λ the system can sustain without becoming unstable. The transfer rate from node i is proportional to the number of particles $N_i(t)$ in the node at time t. Specifically, the rate is given by $\omega_i N_i(t)$, where ω_i is a constant specific to node i.

To determine the stability threshold, we conducted simulations with progressively increasing values of λ, defining the system as unstable only when the number of particles entering node o exceeded the number of particles exiting node d by a factor of 100.

Remarkably, the system remained stable even at high input rates. This stability arises from the proportional rate mechanism, which acts as a self-regulating factor: as the number of particles in a node grows, the transfer rate from that node increases proportionally. This dynamic effectively balances the inflow and outflow of particles across the network, preventing the system from becoming unstable.

Figure 14 illustrates the evolution of the particle count in each node over time for $\lambda = 100$. The results confirm that the proportional rate mechanism maintains stability and prevents the system from experiencing unbounded growth.

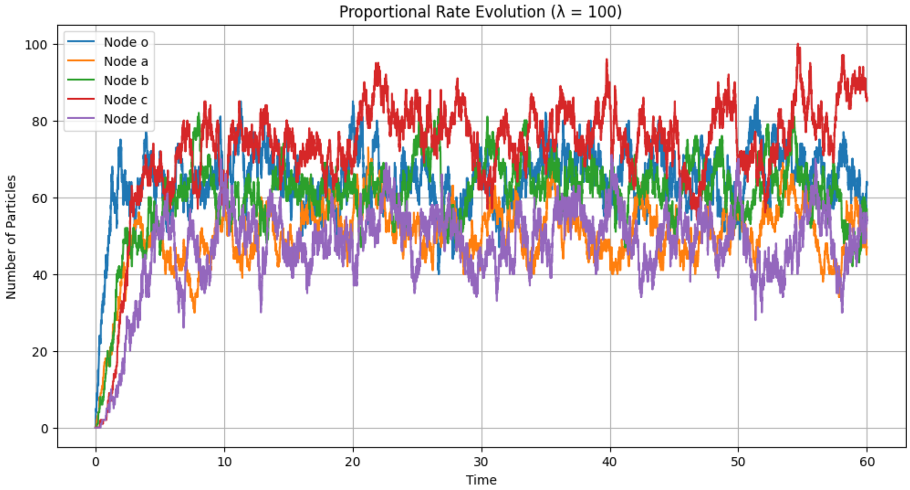

Figure 14: Time evolution of the particle count in each node for the proportional rate scenario with input rate $\lambda = 100$.

b) Fixed Rate

The objective is to simulate the system under the fixed rate scenario for 60 time units, analyze the evolution of the particle distribution over time, and determine the largest input rate λ the system can sustain without becoming unstable. Particles move from node i at a constant rate ω_i, which is independent of the number of particles currently present in the node.

The simulation begins with an input rate $\lambda = 1$, and the evolution of the particle count in each node is tracked over time. Figure 15 illustrates that, for this small input rate, the system remains stable.

To determine the critical threshold for instability, the input rate λ is gradually increased, and the system's behavior is monitored. Unlike the proportional rate scenario, the fixed rate mechanism does not adapt to the particle count in the nodes. As a result, when λ grows beyond a certain limit, the constant transfer rates ω_i cannot balance the incoming particle flow. This imbalance

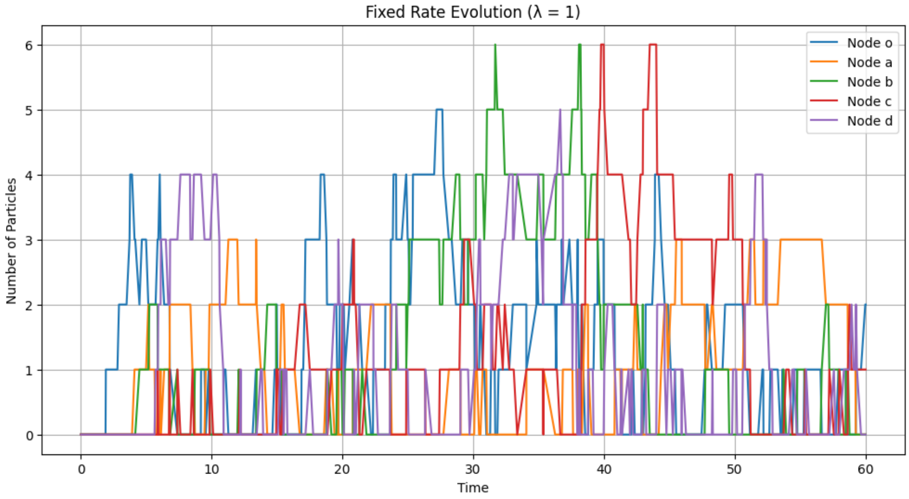

Figure 15: Time evolution of the particle count in each node for the fixed rate scenario with input rate $\lambda = 1$.

causes the particle count to grow unboundedly, as shown in Figure 16.

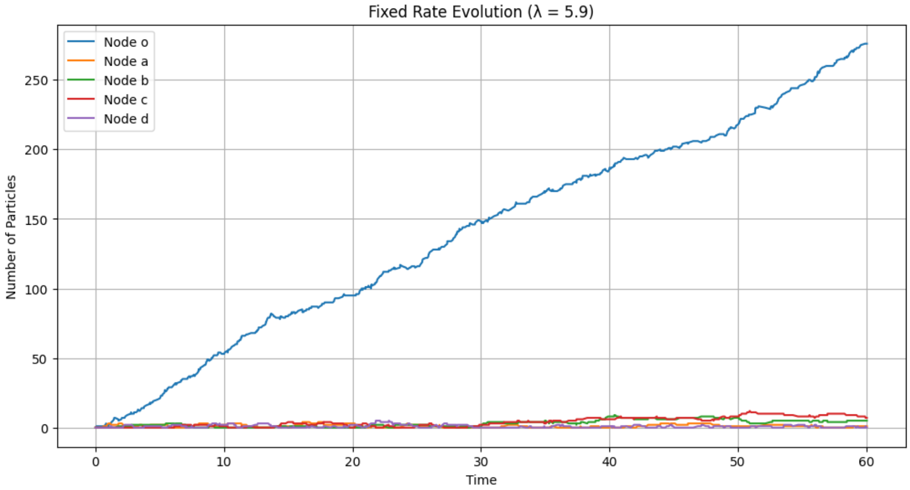

Figure 16: Unbounded growth of the particle count at the critical input rate λ, indicating system instability.

Just for the purpose of the experiment, we defined the system as unstable when the number of particles entering node o exceeds the number of particles leaving node d by a factor of 100. After iterating this process 100 times for various input rates, we determined that the system reaches instability at an average input rate $\lambda \approx 5.9$. This result highlights a fundamental limitation of the fixed rate approach: it lacks a self-regulating mechanism, and for input rates above a critical threshold, the system inevitably becomes unstable.