

Network Dynamics and Learning: Homework 3

Pouria Mohammadalipourahari
Politecnico di Torino
Torino, Italia
s327015@studenti.polito.it

I. PRELIMINARY PARTS

A. Epidemic on a known graph

1) Problem Setup: The task is to simulate an epidemic on a symmetric k -regular undirected graph with node set $\mathcal{V} = \{1, \dots, n\}$ where:

- Each node is directly connected to $k = 6$ closest nodes modulo n
- Total number of nodes $n = 500$
- Simulation runs for 15 weeks
- Initial state: 10 randomly infected nodes

The disease propagation follows a discrete-time simplified SIR model where:

- States: $X_i(t) \in \{S, I, R\}$ (Susceptible, Infected, Recovered)
- Infection probability: $\beta = 0.25$
- Recovery probability: $\rho = 0.6$

The transition probabilities are defined as:

$$P(X_i(t+1) = I | X_i(t) = S, \sum_{j \in V} W_{ij} \delta_{X_j(t)}^I = m) = 1 - (1 - \beta)^m \quad (1)$$

$$P(X_i(t+1) = R | X_i(t) = I) = \rho \quad (2)$$

where m represents the number of infected neighbors for node i .

2) Implementation and Results: We implemented the simulation with $N = 100$ runs to obtain average behavior. Figure 1 shows the results.

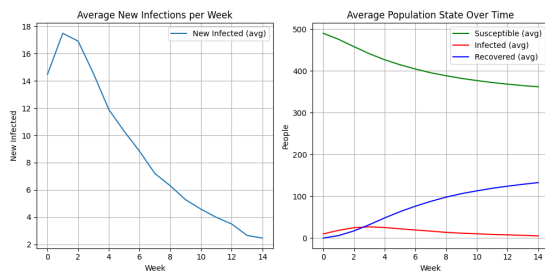


Fig. 1. Simulation results showing (left) average number of new infections per week and (right) evolution of population states over 15 weeks.

3) Analysis: The simulation results reveal several key patterns:

1. New Infections (Left plot):

- Initial phase: Sharp increase reaching peak of 17 new cases per week at week 2

- Middle phase: Steady decline in new cases from weeks 2-8
- Final phase: Gradual approach to zero new cases by week 15

2. Population States (Right plot):

- Susceptible population (green) shows steady decrease, stabilizing at 350 individuals
- Infected population (red) remains relatively low throughout, peaking at 30 individuals
- Recovered population (blue) gradually increases to 150 individuals
- Final state distribution: 70

The regular structure of the graph with $k=6$ connections results in a controlled spread pattern. The high recovery rate ($\rho = 0.6$) combined with moderate infection probability ($\beta = 0.25$) leads to a relatively mild epidemic where most of the population (about 70

B. Generate a random graph

1) Problem Setup: The task is to generate a random graph according to the preferential attachment model. The goal is to have a randomly generated graph with average degree close to k . The process starts with:

- Initial graph G_1 : complete graph with $k + 1$ nodes
- For each time $t \geq 2$, create a new graph $G_t = (V_t, E_t)$ by:
 - Adding a new node to G_{t-1}
 - Connecting it to $c = k/2$ existing nodes
 - Selection probability proportional to node degrees:

$$P(W_{n_t, i}(t) = 1 | G_{t-1}) = \frac{w_i(t-1)}{\sum_{j \in V_{t-1}} w_j(t-1)}, i \in V_{t-1} \quad (3)$$

where:

- Final size: 500 nodes
- Target average degree: $k = 6$

2) Implementation and Results: We implemented the preferential attachment model starting with a complete graph of 7 nodes ($k+1$) and iteratively added nodes until reaching 500. Each new node adds $c = 3$ edges ($k/2$), selecting target nodes with probability proportional to their degrees.

The preferential attachment mechanism leads to a qualitatively different network structure compared to the k -regular graph:

- Some nodes develop high degrees (“hubs”)

- Overall degree distribution follows power law
- Average path lengths typically shorter
- More heterogeneous connectivity

3) Implementation Details: Key aspects of the implementation include:

- Tracking node degrees dynamically
- Preventing multiple edges to same node
- Maintaining connectivity as graph grows
- Handling odd k values by alternating between $\lfloor k/2 \rfloor$ and $\lceil k/2 \rceil$ edges

This graph structure will be used in the next section to study epidemic spread on a more realistic social network topology.

C. Simulate a pandemic without vaccination

1) Problem Setup: The task is to simulate the epidemic on the preferential attachment graph generated in Section B. The simulation parameters are:

- Graph $G = (V, E)$ with $|V| = 500$ nodes and average degree $k = 6$
- Infection probability: $\beta = 0.25$
- Recovery probability: $\rho = 0.6$
- Simulation time: 15 weeks
- Initial state: 10 randomly infected nodes

2) Implementation and Results: Using the methods from Section A, we simulated the epidemic on the preferential attachment graph. The simulation was repeated $N = 100$ times to obtain average behavior. Figure 2 shows the results.

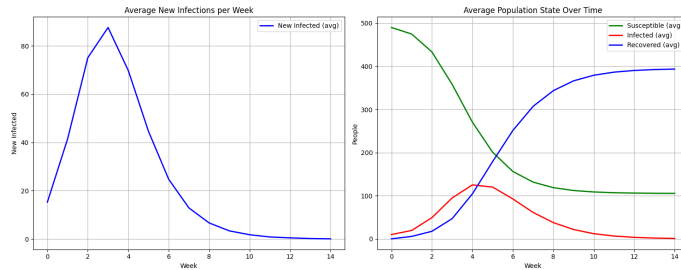


Fig. 2. Simulation results on preferential attachment graph. Left: Average number of new infections peaks at 90 cases in week 3. Right: Population states showing extensive spread with final recovered population 400.

3) Analysis: The simulation reveals significant differences compared to the k -regular graph:

1. New Infections (Left plot):

- Initial phase: Sharp increase to peak of 90 cases at week 3
- Middle phase: Rapid decline from weeks 4-8
- Final phase: Gradual decrease to near zero by week 12-15

2. Population States (Right plot):

- Susceptible population (green) decreases to 100 individuals
- Infected population (red) peaks at 120 individuals around week 5
- Recovered population (blue) steadily increases to 400 individuals
- Final state: 20

The heterogeneous connectivity of the preferential attachment graph produces markedly different epidemic dynamics compared to the k -regular case:

- Much higher peak infection rate (90 vs 17 cases per week)
- Larger total affected population (80)
- More rapid spread and higher peak of infected individuals
- More severe overall impact with majority of population affected

These results demonstrate how network topology significantly influences disease spread patterns, with the heterogeneous preferential attachment network leading to more rapid and extensive transmission. D. Simulate a pandemic with vaccination

1) Problem Setup: In this part, we simulate the pandemic with vaccination strategy. The setup includes:

- Same graph and parameters from Section C:
 - Graph $G = (V, E)$ with $|V| = 500$ nodes and $k = 6$
 - Infection probability: $\beta = 0.25$
 - Recovery probability: $\rho = 0.6$
- Vaccination schedule:
 - Starts at 0% (week 0)
 - Rises to 55% by week 7
 - Maintains 60% through week 15
- Vaccination rules:
 - Vaccinated individuals cannot become infected
 - Vaccination takes effect immediately
 - Random selection from available population

2) Implementation and Results: We implemented the same epidemic model with added vaccination dynamics. Figure 3 shows results averaged over $N = 100$ simulations.

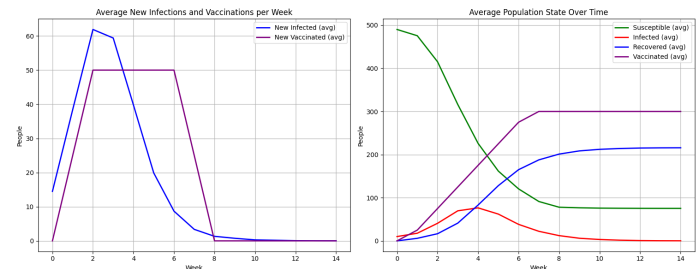


Fig. 3. Left: New infections peak at 60 cases in week 2 while vaccination rate plateaus at 50 per week. Right: Population states showing final distribution among susceptible, recovered, and vaccinated groups.

3) Analysis: The simulation reveals the impact of vaccination:

1. New Infections and Vaccinations (Left plot):

- New infections:
 - Start at 15 cases in week 0
 - Peak at 60 cases in week 2
 - Decline steadily to near zero by week 8
- Vaccination rate:
 - Increases rapidly to 50 people/week by week 2

- Maintains constant rate until week 7
- Drops to zero after reaching target coverage

2. Population States (Right plot):

- Susceptible (green):
 - Starts at 500
 - Decreases steadily
 - Stabilizes at 80 individuals
- Infected (red):
 - Peaks at 70 individuals around week 4
 - Gradually decreases to near zero
- Recovered (blue):
 - Increases steadily
 - Plateaus at 220 individuals
- Vaccinated (purple):
 - Increases linearly until week 7
 - Stabilizes at 300 individuals (60%)

Comparing with no-vaccination scenario:

- Reduced peak infections (60 vs 90 cases per week)
- Smaller recovered population (220 vs 400)
- Better final outcome (60% vaccinated, 40% affected)
- Earlier epidemic containment (week 8 vs week 12)

The vaccination program successfully mitigates the epidemic, despite the preferential attachment network structure that facilitated rapid initial spread.

E. The H1N1 pandemic in Sweden 2009

1) Problem Setup: In this part, we estimate the network structure characteristics and disease-dynamics parameters for the 2009 H1N1 pandemic in Sweden. The setup includes:

- Scaled population: $n = 934$ nodes (scaled by factor of 10^4)
- Historical weekly infection data:
 - $I_0(t) = [1, 1, 3, 5, 9, 17, 32, 32, 17, 5, 2, 1, 0, 0, 0]$
 - First value represents initial number of infected
- Vaccination schedule (percentage of population):
 - $Vacc(t) = [5, 9, 16, 24, 32, 40, 47, 54, 59, 60, 60, 60, 60, 60, 60]$
 - Represents cumulative vaccination coverage
- Parameter search ranges:
 - Initial: $k_0 = 10, \beta_0 = 0.3, \rho_0 = 0.6$
 - Step sizes: $\Delta k = 1, \Delta \beta = 0.1, \Delta \rho = 0.1$

2) Implementation and Results: We implemented a gradient-based parameter search over k , β , and ρ using the RMSE metric:

$$RMSE = \sqrt{\frac{1}{15} \sum_{t=1}^{15} (I(t) - I_0(t))^2} \quad (4)$$

The best parameters found were:

- $k = 12$ (average degree)
- $\beta = 0.2$ (infection probability)
- $\rho = 0.5$ (recovery probability)
- $RMSE = 12.90$

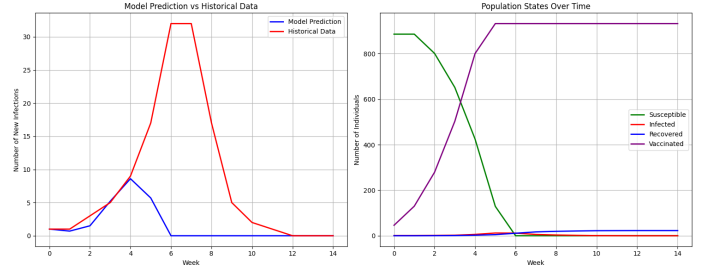


Fig. 4. Left: Comparison between model prediction (blue) and historical data (red) shows model underestimates peak infections. Right: Population states over time showing susceptible (green), infected (red), recovered (blue), and vaccinated (purple) populations.

3) Analysis: The simulation results reveal:

1. Model Prediction vs Historical Data (Left plot):

- Model captures initial growth trend
- Underestimates peak infection (8 vs 32 cases)
- Similar timing of peak (weeks 6-8)
- Consistent decline phase

2. Population States (Right plot):

- Susceptible population decreases to final 30%
- Infected population remains low throughout
- Recovered population reaches 10%
- Vaccinated population achieves target 60%

The estimated parameters suggest:

- Moderate connectivity ($k = 12$)
- Relatively low transmission rate ($\beta = 0.2$)
- Medium recovery period ($\rho = 0.5$)

While the model captures general trends, the discrepancy in peak infections suggests limitations in representing the actual transmission dynamics of the H1N1 pandemic.

II. COLORING

In this part, we study graph coloring as an application of distributed learning in potential games. We examine two cases: a simple line graph and a WiFi channel assignment problem.

A. Simple Line Graph

1) Problem Setup: The task is to color a line graph with 10 nodes where:

- States: $X_i(t) \in \{\text{red, green}\}$
- Initial state: All nodes red
- At each discrete time t , one random node $I(t)$ updates its color
- Update probability is given by:

$$P(X_i(t+1) = a | X(t), I(t) = i) = \frac{e^{-\eta(t) \sum_j W_{ij} c(a, X_j(t))}}{\sum_{s \in C} e^{-\eta(t) \sum_j W_{ij} c(s, X_j(t))}} \quad (5)$$

where:

- Cost function: $c(s, X_j(t)) = \begin{cases} 1 & \text{if } X_j(t) = s \\ 0 & \text{otherwise} \end{cases}$
- Noise parameter: $\eta(t) = t/50$

2) **Implementation and Results:** We ran three simulations starting from all-red configuration. Figure 5 shows the potential function evolution and final coloring.

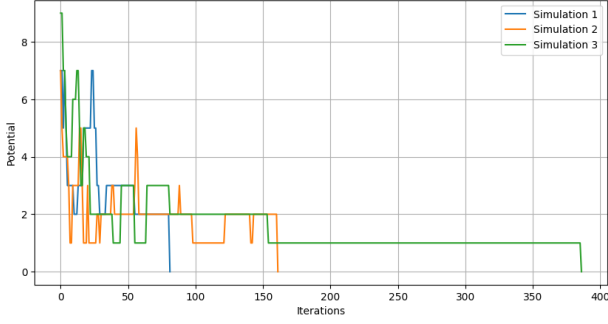


Fig. 5. Potential function evolution over iterations for three simulations, showing convergence to zero (valid coloring).



Fig. 6. Final line graph coloring showing alternating red-green pattern with no adjacent nodes sharing same color.

3) **Analysis:** The simulations demonstrate:

- **Initial Phase (0-50 iterations):**
 - High potential fluctuations
 - Maximum values around 8-9
 - Rapid initial decrease
- **Middle Phase (50-150 iterations):**
 - Smaller fluctuations
 - Potential mostly between 1-3
 - Gradual improvement
- **Final Phase (>150 iterations):**
 - Convergence to zero potential
 - Different convergence times for each simulation
 - Stable alternating color pattern achieved

The algorithm successfully finds a valid two-coloring, with convergence typically occurring within 200-400 iterations.

B. WiFi Channel Assignment Problem

1) **Problem Setup:** The goal is to solve the WiFi channel assignment problem as a graph coloring task. The adjacency matrix and coordinates of the WiFi network are given in `wifi.mat` and `coords.mat`, respectively. The network consists of 100 routers, where:

- Each node represents a router.
- An edge between two nodes indicates potential interference.
- States (channels): $X_i \in \{1 : \text{red}, 2 : \text{green}, 3 : \text{blue}, 4 : \text{yellow}, 5 : \text{magenta}, 6 : \text{cyan}, 7 : \text{white}, 8 : \text{black}\}$.

The cost function is defined as:

$$c(s, X_j) = \begin{cases} 2, & \text{if } X_j = s \text{ (same channel)} \\ 1, & \text{if } |X_j - s| = 1 \text{ (adjacent channel)} \\ 0, & \text{otherwise.} \end{cases}$$

The potential function for the graph is:

$$U(t) = \frac{1}{2} \sum_{i,j \in V} W_{ij} c(X_i(t), X_j(t)),$$

where W_{ij} is the adjacency matrix.

The algorithm starts with a random channel assignment and updates the state of one node at a time based on a probability distribution given by the softmax function:

$$P(X_i(t+1) = s) = \frac{e^{-\eta(t) \sum_j W_{ij} c(s, X_j(t))}}{\sum_k e^{-\eta(t) \sum_j W_{ij} c(k, X_j(t))}},$$

where $\eta(t) = \frac{t}{50}$ is the inverse noise parameter.

Objectives:

- Visualize the initial and final configurations of the graph.
- Track the potential function over iterations for convergence.
- Test different values of η and analyze the results.

2) Implementation and Results:

Initial Random Assignment: Figure 7 shows the graph's initial configuration with random channel assignments. Each color corresponds to a channel.

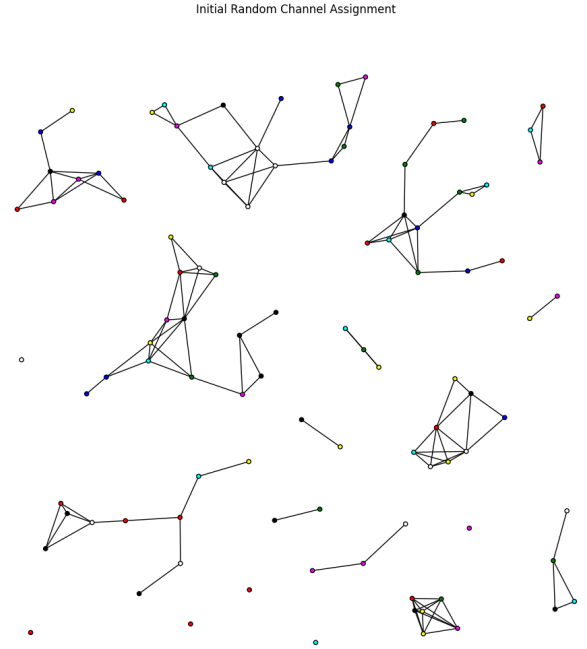


Fig. 7. Initial random channel assignment.

Convergence of Potential Function: Figure 8 shows the evolution of the potential function over iterations for three independent simulations. The algorithm converges to a near-zero potential, indicating a valid channel assignment.

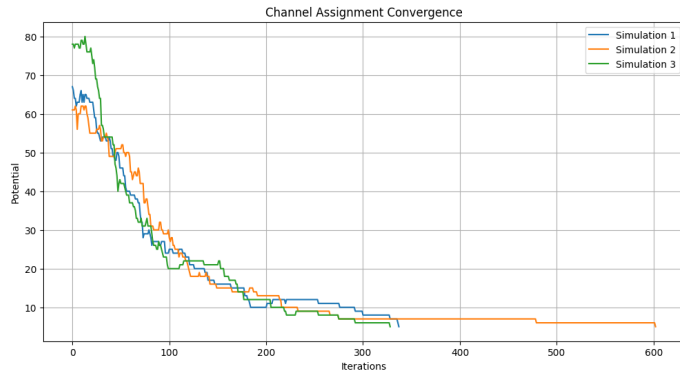


Fig. 8. Potential function convergence for three simulations.

Final Channel Assignment: Figure 9 illustrates the final graph configuration with minimized potential. The coloring ensures minimal interference.

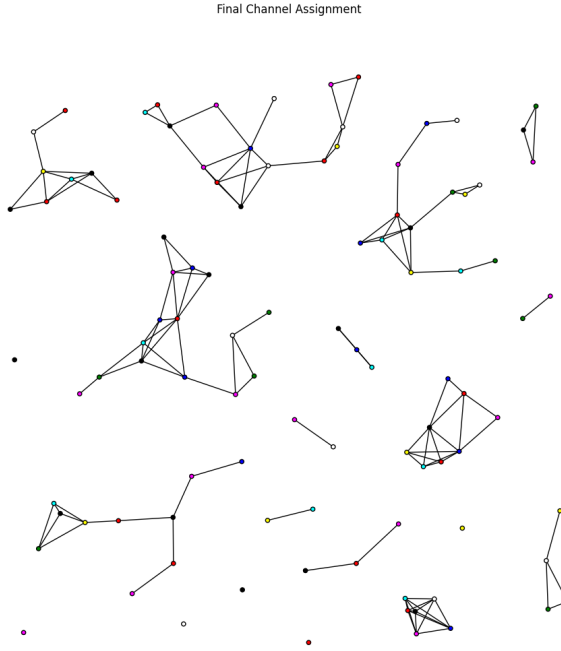


Fig. 9. Final channel assignment.

Impact of Different η Values: Figure 10 compares the potential function's convergence for different fixed values of η ($\eta = 10^{-5}$, $\eta = 1$, and $\eta = 100$). The choice of η significantly affects convergence speed and stability.

3) Analysis:

- The algorithm successfully minimizes the potential function to near-zero, indicating an interference-free channel assignment.
- Increasing η accelerates convergence but may lead to instability for excessively large values.
- The distributed nature of the algorithm makes it scalable for larger networks.

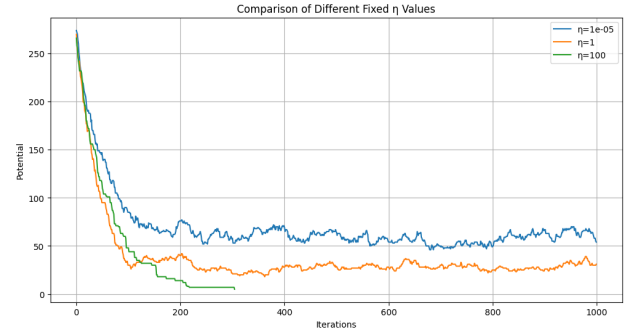


Fig. 10. Comparison of potential function convergence for different η values ($\eta = 10^{-5}$, $\eta = 1$, $\eta = 100$).