Project Report: Simulating Infectious Disease Spread in a Social Network

Muhammad Zain Abbas (505230)
Absar Ahmad (519258)
Muhammad Hassaan Nadeem (506382)

Course: Data Structures and Algorithms
Course Instructor: Sir Aimal Rextin

Contents

1 Description and Assumptions

This project is focused on simulating disease transmission in a community. While graph theory has made it quite easy to simulate infectious diseases and their spread in a community, existing models often lack closeness to reality. What sets our project apart is the focus on getting as close to reality as one can. To construct a graph-based simulation that closely mirrors real-world town dynamics, we established several foundational assumptions regarding the environment and the pathogen.

1.1 The Environment

We model a closed-system township. Apart from people, we have implemented community hubs like parks, markets, schools, and offices to simulate reality. The town operates on a circadian schedule (Morning, Evening, Night), which drives the topological changes in our network.

1.2 The Pathogen

The disease is modeled as a respiratory virus with probabilistic transmission. We assume transmission is not binary but follows a continuous probability distribution. The disease can spread through relations (friends, family, etc.) and through community hubs which people often visit. The hub can get contaminated, and the virus can linger in a Hub Node even after the infectious Agent Node has disconnected.

2 Human Node Architecture

In our model, individuals are not simple data points but complex **Nodes** containing detailed internal states. To achieve a high degree of realism, we classified these nodes into specific demographic classes, each possessing unique properties that influence the graph's behavior.

2.1 Demographic Classes

- **Child Nodes:** These represent the children in a community. They go to school and possess high asymptomatic spread potential and low mortality rates.

- **Adult Nodes:** They represent the adults in a community, who have to go to work in offices or factories daily. Their behavior is marked as high mobility, so they often go to markets and cafes as well.

- **Senior Nodes:** These are the aged people in a community; their health is not very strong, so they have high susceptibility and mortality coefficients.

2.2   Node Attributes

Each Node encapsulates distinct attributes like **age, assigned home-id, vaccination status, immunity, is-superspreader**, etc. Other important nodes which require explanation are given below:

- **Constitution (Natural Disease Resistance)** ($C \in [0, 1]$)**:** A stochastic value determining resistance to infection based on their age and how well-maintained their health is.

- **State Machine:** Tracks the epidemiological status. SEIRD model but with added states like asymptomatic, symptomatic, recovered, and vaccinated.

- **Affiliation Pointers:** Permanent references to Structural Hubs like schools, offices, factories, etc.

- **Mobility Type:** Low, Moderate, High. It dictates how likely the people are to go to cafes, parks, markets, and other hubs. People with low mobility often tend to stay at home (introverts) and vice-versa.

- **Immunity:** After a person recovers from a disease, they have high immunity which fades with the passage of time.

3   Graph Topology and Community Structure

To represent the complex interaction between people and places, we designed the simulation as a **Bipartite Graph** ($G = (U, V, E)$) with two distinct types of nodes.

- **Set U (Agents):** The population of the town.

- **Set V (Hubs):** Locations including Households, Schools, Workplaces, Markets, Hospitals, and Cemeteries.

Figure 1: Graph Visualization

3.1 Static Structural Layer

The "backbone" of our graph is formed by static edges connecting Agent Nodes to their assigned Household Nodes.

- **Cluster Formation:** By assigning multiple agents to a single Household Node and applying attractive spring forces, the graph naturally organizes into tight "family clusters" on the periphery.

- **Institutional Links:** Agents also possess static edges to their Workplaces or Schools, creating a secondary layer of community organization that spans across different families.

4 Temporal Dynamics and Edge Weights

A static graph cannot capture the fluidity of human movement. We implemented a **Temporal Network** where the **existence** of edges is constant, but their **influence** (Weight) changes dynamically over time (Morning, Evening, Night).

4.1 The Weighted Flow Model

Instead of calculating coordinate geometry for every interaction, we utilize **Edge Weights** ($w \in [0,1]$) to represent presence.

- **Morning Phase:** The system increases the weight of edges connecting Child Nodes to School Nodes ($w \to 1.0$) and decreases the weight of edges to Household Nodes ($w \to 0.0$).

- **Evening Phase:** Based on probabilistic routines, Adult Nodes increase the weight of edges connecting to Markets or Parks.

This approach allows us to model "going to work" or "visiting the market" purely as graph operations. The visualization represents this by fading edge colors in and out, showing the "flow" of the population without disrupting the topological stability of the graph.

5 Graph-Based Disease Transmission

The transmission of the virus is the primary function of the graph. We calculate infection probability using two distinct graph traversal methods, governed by global parameters **Beta (β)** (Infectivity) and **Kappa (κ)** (Spatial Range).

The probability of infection P over time step Δt is given by:

$$P_{inf} = 1 - e^{(-\lambda \cdot \Delta t)} \tag{1}$$

Where λ (Force of Infection) is derived from the following graph interactions:

5.1 Node Attribute-Dependent Edge Weighting

A core challenge in simulating disease spread on a static graph is the "Teleportation Problem." In a standard graph traversal, if Node A is connected to Node B, transmission is often treated as a constant probability. However, in a real-world town, a "Friendship Edge" does not imply constant physical contact.

To address this, we implemented a **Spatially-Embedded Network Model**. In our architecture, the Graph Topology defines the *possibility* of interaction, while the Spatial Geometry determines the *intensity* of that interaction.
Every Agent Node u in our graph $G = (V, E)$ possesses a dynamic state attribute $P_u(t) = (x, y)$, representing its coordinates in the simulation space at time t.

While the set of edges E (social connections) remains static, the effective **transmission weight** W_{uv} of an edge between nodes u and v is recalculated at every discrete time step based on their dynamic attributes. The force of infection λ along a social edge is given by:

$$\lambda_{social}(u, v, t) = \underbrace{A_{uv}}_{\text{Topology}} \times \underbrace{\beta}_{\text{Infectivity}} \times \underbrace{\left(\frac{1}{1 + \left(\frac{||P_u(t) - P_v(t)||}{\kappa} \right)^2} \right)}_{\text{Spatial Decay Function}} \tag{2}$$

Where:

- A_{uv} is the Adjacency Matrix value (1 if edge exists, 0 if not). This acts as the strict **Topological Filter**. Even if two nodes share the same spatial coordinate, transmission is impossible if $A_{uv} = 0$.

- $||P_u(t) - P_v(t)||$ is the Euclidean distance between the nodes' current dynamic positions.

- κ (Kappa) is the spatial decay constant, modulating the effective range of the pathogen.

Justification of Design: This mechanism validates our model as a **Dynamic Graph** Simulation. By treating spatial position as a time-variant attribute of the Node, we achieve two critical "Close-to-Reality" behaviors:

1. **Prevention of Non-Local Spread:** An infected agent at a "Market" Hub cannot infect a connected friend who is currently at a "Workplace" Hub, despite the existence of a graph edge. The large Euclidean distance drives the edge weight $\rightarrow 0$.

2. **Co-Location Activation:** The edge effectively "activates" only when both nodes traverse to similar coordinates (e.g., both return to the same Household Node at night).

Thus, the disease does not spread purely spatially (which would ignore social circles), nor purely topologically (which ignores physics), but through the intersection of both, strictly governed by the graph's Adjacency Matrix.

5.2 Hub-Based Transmission (Node Accumulation)

This models environmental spread. Hub Nodes act as accumulators in the graph.

- **Upload:** Infectious nodes push viral load to connected Hub Nodes weighted by the edge strength (w).

- **Accumulation:** The Hub Node stores this value, representing aerosol concentration.

- **Download:** Susceptible nodes pull risk from the Hub Node.

$$\lambda_{hub} = \beta \cdot w \cdot \left(\frac{\sum \text{ViralLoad}}{\text{Capacity}_{hub}} \right) \tag{3}$$

6 Visualization Strategy

Our visualization is designed to render the underlying graph structure explicitly. We developed a **custom Force-Directed Graph** algorithm to make the visualization exactly like what we wanted. We have placed the community hubs like Market, park, schools, and hospitals at the center to represent the town center. The humans are shown around these hubs in a ring (yes, we utilized a ring to visualize our nodes better). And the cemetery and hospital are outside this people ring. The visualization of the edges is described below.

- **Structural Web:** Faint, static edges represent the permanent family and social bonds (the "resting state" of the community).

- **Dynamic Flows:** As edge weights increase (e.g., an agent going to the Market), the edge is rendered with increasing opacity and distinct colors (Orange for Markets, Green for Parks, etc.). This visualizes the temporal "pulse" of the town.

- **Transmission Vectors:** When infection occurs, the specific edge responsible for the transmission flashes Red, visualizing the exact path of the virus.

- **Other Visualization:** Edges connecting Hospitalized Nodes to the Hospital turn magenta in color. Also, the nodes are connected to the cemetery to visualize the death toll.
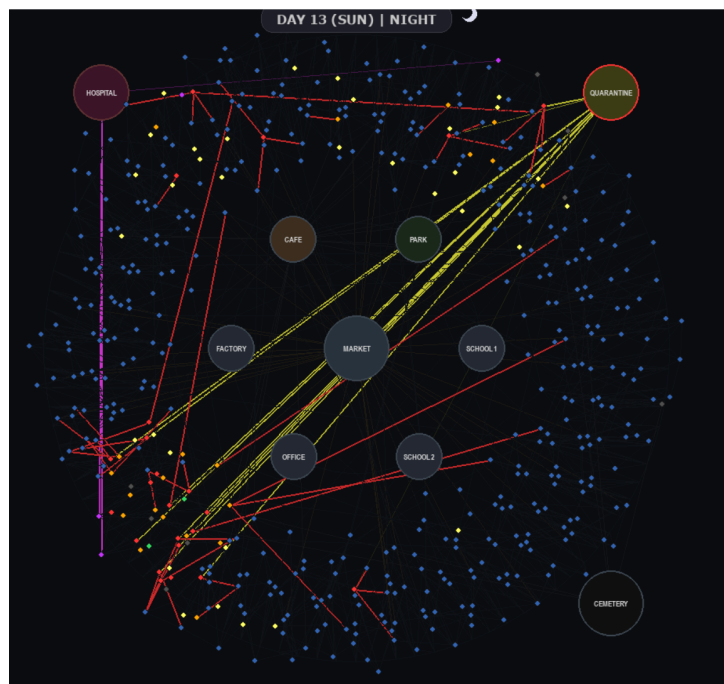
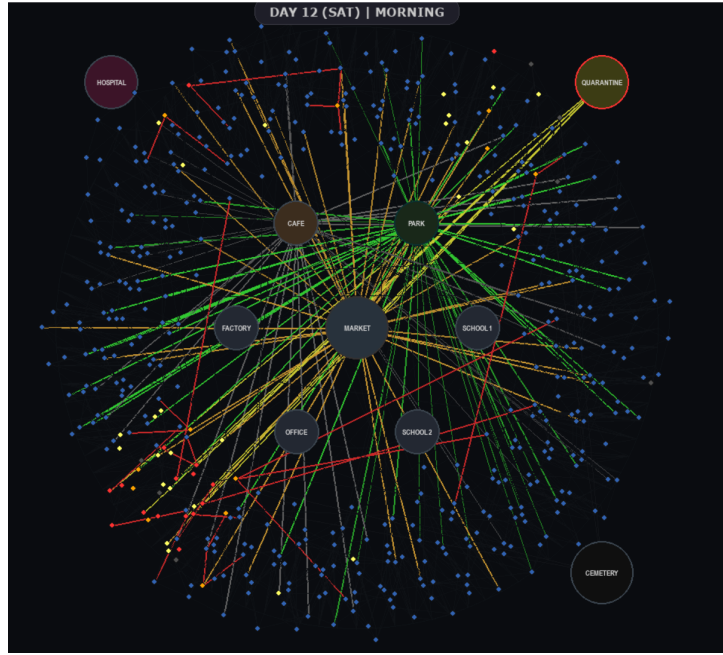

Figure 2: Graph Visualization

Figure 3: Graph Visualization during the evening, the nodes connecting to hubs have gained color which represent their weight i.e. that these nodes are visiting them.

Spatial Simulation Mode: We implemented an additional visualization mode which dynamically represents the movement of the nodes to the community hubs and their daily movements based on the time. We were very excited about this since this looks very close to reality, as if you are watching an actual town and its people moving, sitting in the sky somewhere in the clouds and seeing the disease spread in real-time.
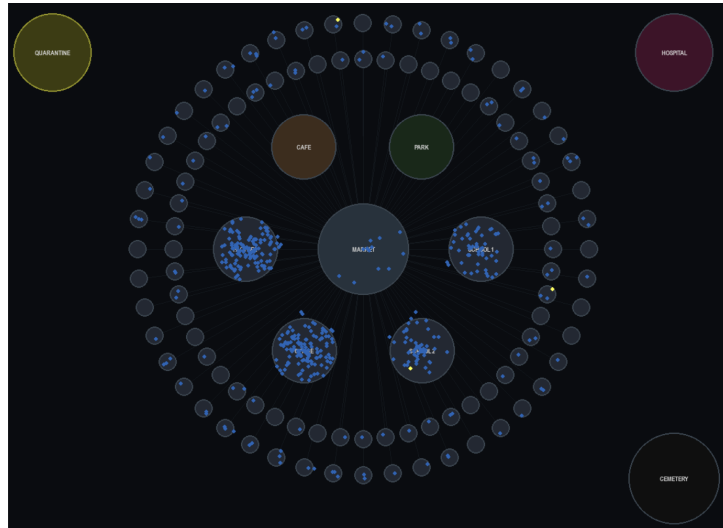

Figure 4: Spatial Simulation during the Morning, most of the nodes are at work, school or factory with few at home and market
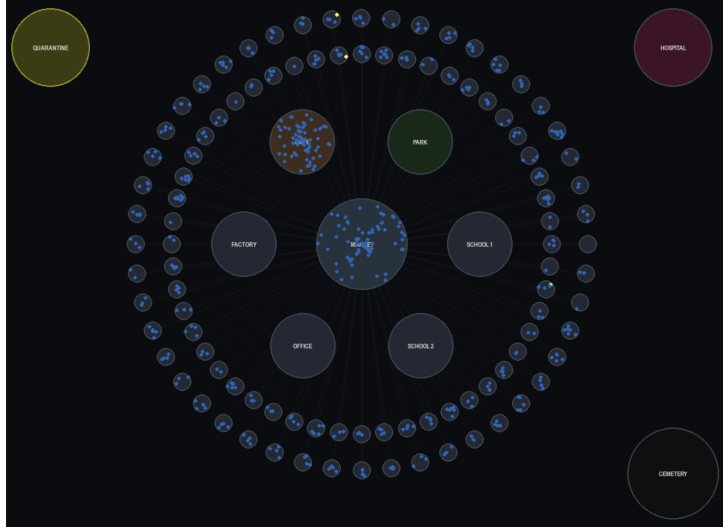
9

Figure 5: Spatial Simulation during the Evening, nodes visit parks and cafe along with market

7 Algorithmic Design Choices

7.1 Graph Generation Model

We selected the **Watts-Strogatz Small-World Model** for generating the social graph.

- **Why not Erdős–Rényi (Random Graph)?** Random graphs lack local clustering. In reality, for example, two friends of a person are likely to be friends with each other. Random graphs fail to capture this essential social structure.

- **Why not Barabási–Albert (Scale-Free)?** Scale-free networks are dominated by massive hubs (influencers). While applicable to the internet, physical town interactions are constrained by geography and time. A regular citizen does not have 10,000 physical contacts.

- **Watts-Strogatz Choice:** This model provides high clustering coefficients (families/circles) while allowing for short average path lengths (the "six degrees of separation" phenomenon; yes, we got this idea from the video of the Veritasium channel on YouTube and it is very interesting actually), which is ideal for modeling disease spread in a town.

7.2 Complexity Optimization: Spatial Hashing

A standard Force-Directed layout requires calculating repulsion forces between every pair of nodes, leading to a time complexity of $O(N^2)$. For $N = 500$ agents, this results in 250,000 calculations per frame, causing significant lag.

Our Solution: We implemented a **Spatial Grid** (Binning) optimization. Here is how we tend to explain it:

1. The simulation space is divided into a grid of buckets.

2. Nodes are hashed into these buckets based on coordinates.

3. Repulsion is only calculated against nodes in the same or adjacent buckets.

This reduces the effective complexity to $O(N \cdot k)$ (where k is local density), allowing for a smooth 60 FPS experience.

8   Advanced Simulation Factors

To elevate the model beyond a theoretical abstraction and approximate the stochastic nature of real-world epidemiology, we integrated a comprehensive set of agent-specific and environmental variables:

- **Demographic Heterogeneity:** The graph is not composed of identical nodes. Agents are classified into **Child**, **Adult**, and **Senior** classes. This determines their graph connectivity (School vs. Work edges) and influences their *Constitution* attribute, a stochastic value $(0.0 - 1.0)$ that modulates resistance to infection and mortality risk.

- **Asymptomatic and Presymptomatic Transmission:** We split the infectious state into **Symptomatic** (I_s) and **Asymptomatic** (I_a). Asymptomatic nodes act as "Silent Spreaders," traversing the graph and forming dynamic edges without being restricted to the Household node (unlike Symptomatic nodes which self-isolate), thereby increasing the effective reproduction number (R_t).

- **Super-Spreader Nodes:** A randomized subset of the population (5%) is assigned the *Super-spreader* attribute. These nodes possess a viral shedding magnitude $3\times$ higher than average, allowing them to disproportionately influence the infection rate of connected Hubs.

- **Dynamic Viral Load Curves:** Infection is not modeled as a binary "On/Off" state. We implemented a temporal function where viral shedding ramps up, peaks, and decays over the infection duration. This ensures that the "Force of Infection" transmitted along an edge varies realistically based on the source node's days-since-exposure.

- **Vaccination State:** A boolean node attribute that reduces susceptibility by a factor of ν (Vaccine Efficacy) and critically reduces viral shedding output if a breakthrough infection occurs, simulating the dampening effect of herd immunity.

- **Immunity Waning (SEIRS Model):** Recovered nodes do not remain in state R indefinitely. Their immunity coefficient follows an exponential decay function over time ($e^{-\lambda t}$), eventually transitioning them back to state S, allowing the simulation to model re-infection waves and endemic equilibrium.

- **Lockdown Protocols:** A global system parameter that strictly sets the edge weights of non-essential dynamic Hubs (Parks, Cafes, Markets) to 0.0. This effectively alters the graph topology in real-time, severing specific high-degree connections to flatten the infection curve.

- **Quarantine Center:** We added a separate quarantine center where the infected people are sent to isolate them from the community. This is another real-life measure taken against contagious diseases that we have implemented in our simulation.

- **Hospital Capacity Dynamics:** The Hospital Hub has a fixed capacity. If the number of hospitalized nodes exceeds this threshold, the mortality probability function for critical nodes receives a steep penalty multiplier, simulating the collapse of healthcare infrastructure.

9 Results and Observations

We ran our simulation by tweaking various variables that we have given in the UI. Now there can be multiple end-results of the simulation based on the values of these variables, which would be too much to explain over here. We, however, observed some very interesting results, which we are very excited to state in here:

1. **Late-stage Lockdown:** Now this is one of the most interesting observations. If the lockdown is implemented in a town in the later stages, it has close to no effect, since sufficient nodes have already become infectious and they will keep spreading the disease to their family members even in their houses, specifically if they are asymptomatic. This result has been observed in real life as expected as well.

2. **Early-stage Lockdown:** If the lockdown is implemented in the early stages of disease spread, it only tends to slow down the disease spread, but nodes keep getting infected gradually. The reason for this in our simulation is that we allowed some nodes to still travel outside to represent the necessary movements like people still need to buy groceries in order to live and the essential workers like paramedical staff, police, gas, and electricity workers, etc. This, however, ensures that the hospital capacity is not overloaded and slightly reduces the mortality rate.

3. **Vaccination:** Vaccination proved to be the most effective way to end the disease spread in our simulation. Vaccination tends to flatten the curve of the infected people and eventually bring it down.

4. **Quarantine:** Quarantine is another effective strategy to effectively slow down the transmission of the disease. However, it often needs to be paired up with lockdown and vaccination to be very effective.

5. **Mask Compliance:** Similar to lockdown, mask compliance without lockdown or vaccination only tends to slightly slow down the transmission, but nodes still keep getting infected. This result is also observed in real life.

Best Strategy to Stop Disease Spread: Thus from the simulations, the best strategy to stop disease spread is to implement an early-stage lockdown and quarantine, with strict mask compliance to ensure minimal spread for the necessary outgoing. This will slow down the disease spread. In the meanwhile, vaccination must be started to end the disease spread, and the hospital capacity must be increased to reduce the mortality rate.
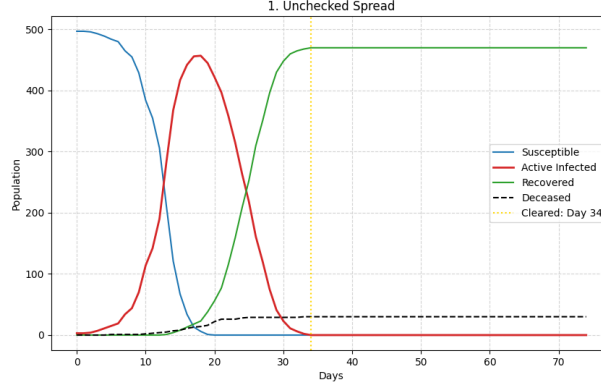

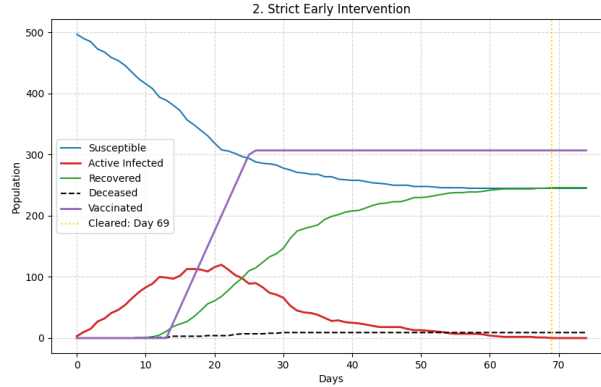Figure 6: Simulation Result with no preventions, lockdown, quarantine or vaccination

Figure 7: Simulation result with early lockdown at 20 cases and start of vaccination along with quarantine, mask and other measures along with that

10   Conclusion

This project successfully demonstrates the power of Graph Theory in simulating complex biological and social phenomena. By utilizing a Temporal Bipartite Graph, we were able to model the nuanced interaction between static social structures and dynamic daily routines. The implementation of Weighted Edge Flows for transmission, combined with $O(N)$ physics optimizations, resulted in a simulation that is both computationally efficient and highly representative of real-world epidemiological dynamics.