IDEOLOGY SPREAD MODELING IN A POPULATION

Saksham Panghal Pranav Kala Anarghya Das Anjaneya Damle

1. Abstract

In this project, we have explored how political opinions / ideologies spread in a population, and how these spreading opinions result in polarised communities. We have made the use of Graphs, KD-Trees and Union-Find, along with required algorithms which shall be discussed in the report. Finally, we have prepared a visualisation of the simulation of our work, which combines graph generation using the Barabási–Albert model, and opinion-update dynamics inspired by the Friedkin–Johnsen (FJ) model.

2. Problem Statement

In real-world social systems, individuals often form communities based on shared opinions, leading to phenomena such as ideological clusters, echo chambers, opinion polarization, and even physical or digital “ghettos.” Identifying such communities is important for understanding collective behaviour, predicting polarization trends, and analysing how information or influence spreads across society.
To study these processes, it is essential to develop an accurate model of how opinions propagate within a network so that we can approximate the distribution and evolution of opinions in a real-life population. This project aims to construct such a simulation framework and use it to analyse the formation and fragmentation of opinion-based communities.

3. Proposed Solution & Analysis

Graph Modelling:

To begin with, we first need to model a random graph. However, since a purely random graph may lead to opinion neutralization in the entire graph, we used the Barabasi Albert model, modified to our use case with a spatial variant. What this does is that, initially there shall only be a set of few people who have their own opinions, and the rest of the graph is unopinionated. This represents the stage when an ideology is kindled. Eventually, others start forming opinions of the same and the opinions start to spread. Also, since in real life people are more likely to be influenced by nearby people, we make a modification of our own for a more realistic modelling.

The standard Barabasi-Albert model generates a scale-free network by:

1. Start with small fully connected graph of m_0 nodes.
2. Add new nodes one by one
3. Each new node connects to m existing nodes with probability

$$P(i) = \frac{k_i}{\sum_j k_j}$$

where k_i is the degree of node i

As a consequence of this model, high-degree nodes attract more links (rich get richer), and hubs form.

In our modification, if a node is unable to find any links to connect to, it searches for nearby neighbours, leading to localised clusters:

When a new node i arrives (for $i = 1, \dots, N-1$):

1. Define the spatial neighbourhood

$$\mathcal{N}_i = \{\, j \in \{0, \dots, i-1\} : \,||\mathbf{x_i} - \mathbf{x_j}||_2 \leq R \,\}.$$

where R is the hyperparameter connection radius.

2. If $\mathcal{N}_i = \emptyset$, then take the K nearest historical nodes, i.e.

$$\mathcal{M}_i = argmin_K[\, ||\mathbf{x_i} - \mathbf{x_j}||_2 \colon j \in \{0, \dots, i-1\} \,]$$

To optimize searching for the K nearest nodes, we use KD-Trees, which optimizes the time complexity from $O(N)$ to $O(logN + K)$, for each new node i.

Opinion Dynamics:

To simulate the spread of opinions, we use a modified version of the Friedkin-Johnsen Model:

$$x_i(t+1) = s_i(t) \cdot x_i(t) + \left(1 - s_i(t)\right) \cdot \left[\phi_i(t) \cdot \bar{x}_i(t) + \left(1 - \phi_i(t)\right) \cdot x_i(t)\right]$$

Where:

$x_i(t) \in$ [-1, 1] : Opinion of agent $x_i(t)$ at time t

$s_i(t)$: Stubbornness (resistance to change) of agent i

$\bar{x}_i(t)$: Weighted average opinion of neighbours

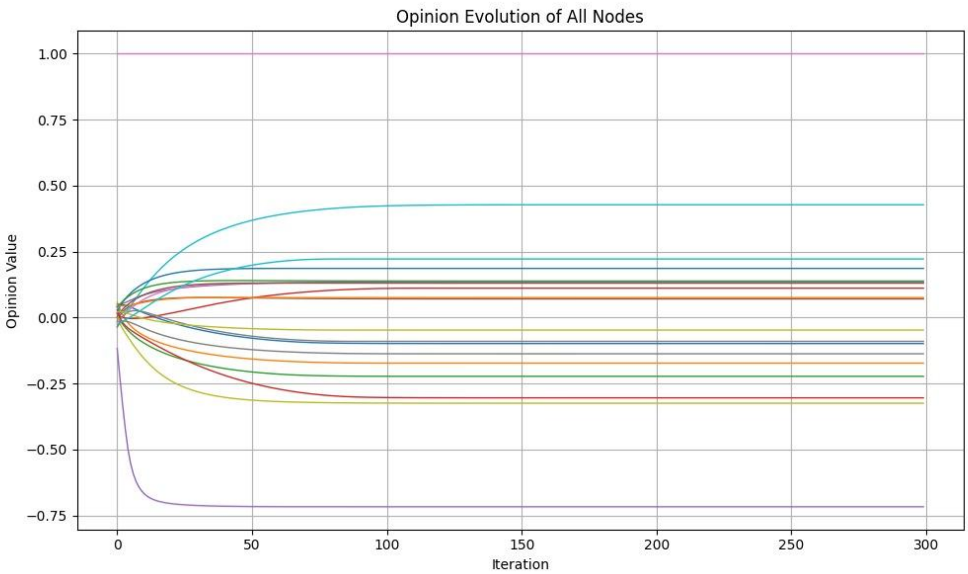
$\phi_i(t)$: Confidence factor based on opinion distance

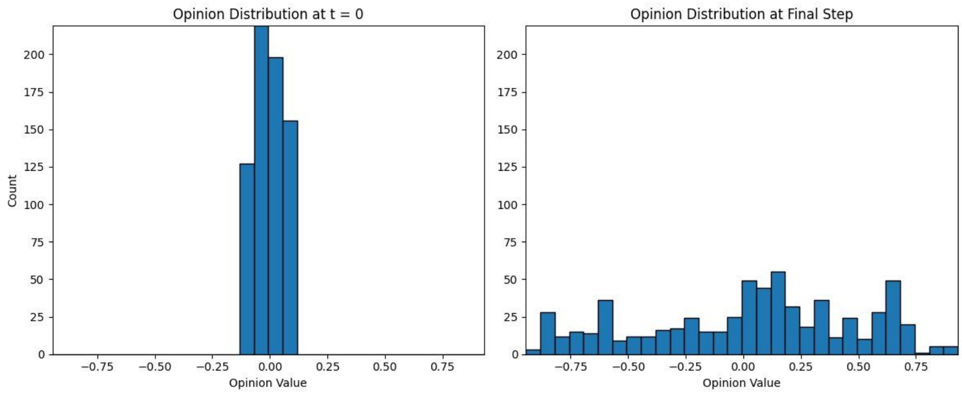
where:

$$\bar{x}_i(t) = \begin{cases} \dfrac{\sum_{j \epsilon \mathcal{N}_i} w_j . x_j(t)}{\sum_{j \epsilon \mathcal{N}_i} w_j}, & \displaystyle\sum_{j \epsilon \mathcal{N}_i} w_j . > 0 \\ x_i(t), & otherwise \end{cases}$$

Since over time, people become more rigid in their opinions, we added a stubbornness factor which increase with time to simulate the same. As a consequence, the modelled graph stabilises eventually.

$$s_i(t+1) = \min(1,\, s_i(t) + \delta_s)$$





Community Detection

After stabilisation, we observe that people who have similar opinions form communities. We used Union-Find to detect & highlight these communities. We make Disjoint sets where each disjoint set represents a discovered ghetto. Groups of people who are spatially close and have very similar opinions fall into the same community.

DSU merges (unites) nodes that satisfy these conditions –

1. Have the same opinion sign (positive / negative / neutral),
2. Have very similar exact opinion value,
3. Are physically close in space (within MAX_RADIUS distance).

4. Data Structures Used

a. Graphs

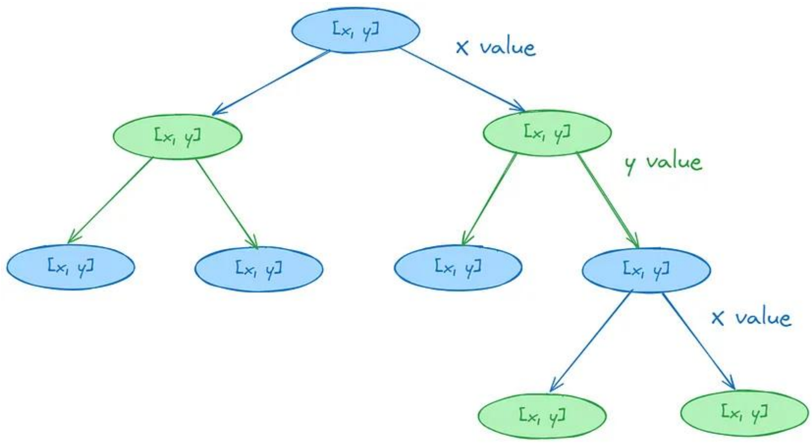
We have modelled the population into a graph, where each node represents a person. Depending on the opinion, the nodes have been assigned three colours – red, blue or grey. The size of each node differs based on the magnitude of opinion each person has.

b. KD-Trees

When searching for nearest neighbouring nodes while adding nodes during graph generation, iteratively searching for nearest nodes results in an inefficient time complexity of $O(n)$ per node. To optimise this, we made use of KD-Trees, which does this in $O(\log n)$. How this achieves this is as follows:

A KD-Tree organizes k-dimensional (2D in our case) points by recursively splitting the space with axis-aligned hyperplanes. At each level, the tree chooses one coordinate axis (cycling x → y → x → …), selects the median point on that axis, and divides the remaining points into left (smaller coordinate) and right (larger coordinate) subtrees. This produces a balanced binary tree where each node corresponds to a progressively smaller region of space.

For nearest-neighbour search, the query point is first pushed down the tree to a leaf to obtain an initial candidate. The algorithm then backtracks, checking whether the hypersphere defined by the current best distance intersects the sibling regions. If not, those regions are safely pruned; if yes, they are explored. Because large parts of the space get eliminated by these axis-based partitions, only a small number of nodes must be visited.



c. Disjoint-Set

We used a Disjoint Set Union structure to identify communities of people who form tightly connected and ideologically similar groups. After the opinion dynamics stabilize, the DSU merges pairs of individuals who are spatially close and share nearly the same opinion. By repeatedly uniting such pairs, the DSU efficiently discovers “ghettos” or echo-chambers, which are connected components of like-minded individuals. This allows us to detect polarized pockets in the population without repeatedly running costly graph traversal algorithms.