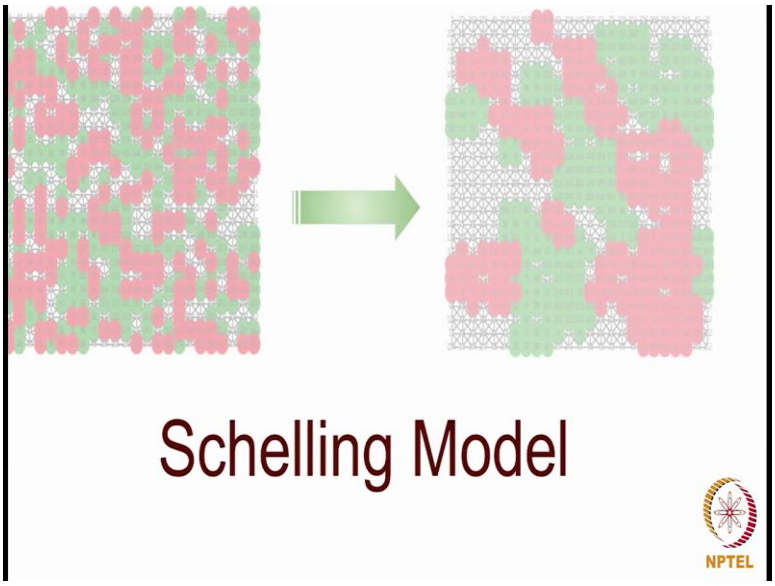
Social Networks
Prof. S. R. S. Iyengar
Department of Computer Science
Indian Institute of Technology, Ropar

Lecture – 57
Homophily (Continued) & Positive and Negative Relationships
Schelling Model Implementation – Introduction

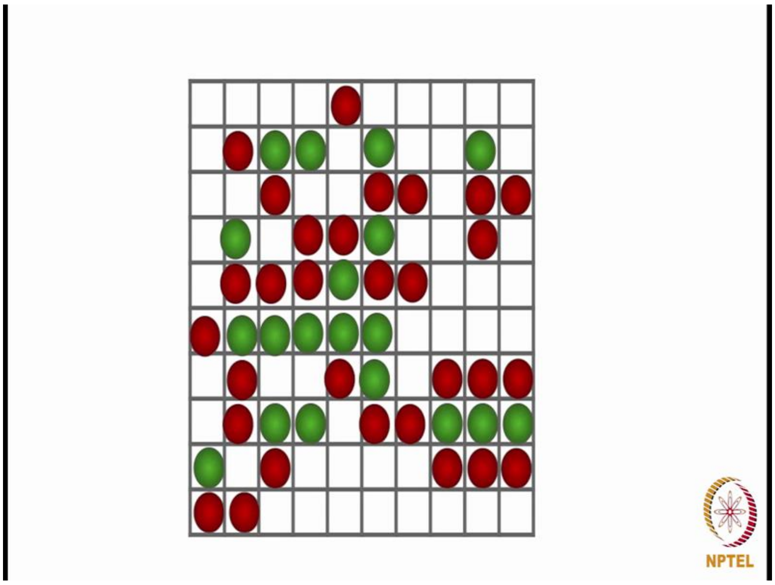
(Refer Slide Time: 00:05)



Schelling Model

Hey everyone, Schelling model shows how the effect of homophily that operates at a local level leads to some interesting global patterns in the network. In this video we are going to implement this model and we will observe the patterns that emerge. Before we start the implementation I will give you a brief of the model, let us assume that there is a population of people and these people are one of the 2 types that is every person is either of type 0 or type 1.

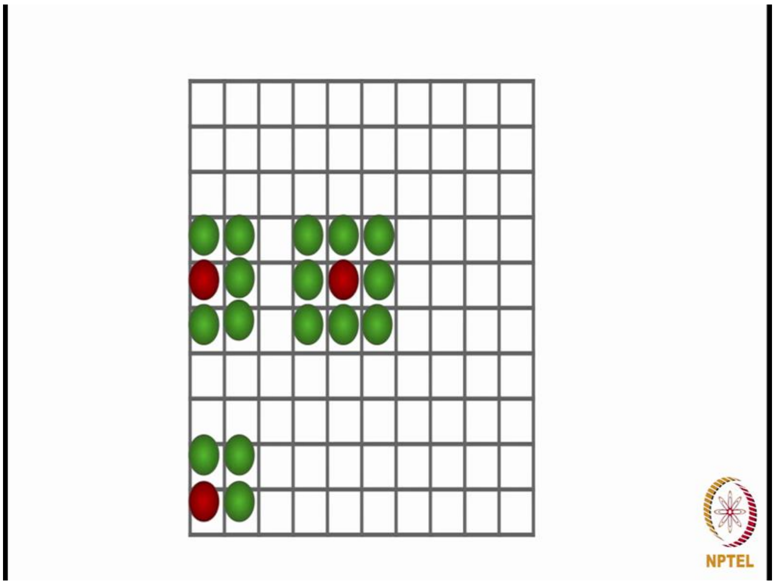
(Refer Slide Time: 00:44)



Further assume that these people live in a city which is like a grid that is there are cells in this grid where these people live just like this. Also, some cells of this grid have people whereas, some cells will be empty, in other words every cells cell will either have a person of type 0 or a person of type 1 or it will be empty. For example, like this as you can see every cell either has a red node or you can say red person or a green person or it is empty, correct.

Now, cells neighbours are the cells that touch it including the diagonal cells. So, you can imagine that the cell that is not on the boundary will have 8 neighbours.

(Refer Slide Time: 01:24)

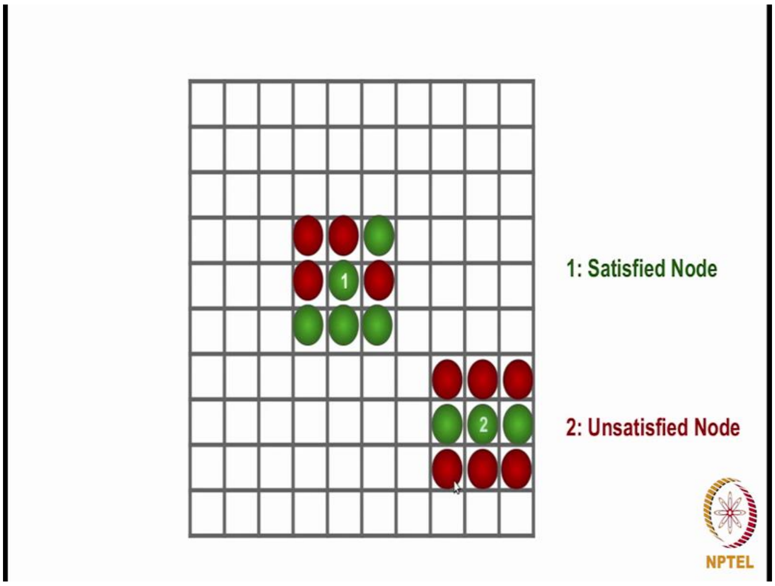


For example, if you look at this scenario this red node is an internal node that is not on the boundary. So, every node which is not on the boundary will have 8 neighbours like this the red person has basically the cell that has red person has 8 cells which are its neighbouring cells. Similarly, if you look at the boundary cells they can be 2 cases first case could be where the cell is a corner cell like here the red person is staying in a cell which is a corner cell. So, this cell has 3 neighbours as you can see here the other kinds of boundary cells can be like this which are not the corners.

So, as you can see this red person which is residing in the cell has five neighbours I think it is not difficult to understand that we can convert this whole scenario into a graph where the cells are the nodes and there is an edge between 2 cells if they are neighbours on this grid. So, that is what we are going to do we are going to analyze this grid using a network now if you look at a metropolitan city where there are people from different states of the country everyone tends to or rather prefers to stay in an area where there are more people from their own state. So, that practically happens we are going to apply the same scenario in our given grid. So, the people in our model wish to have at least some other people of their own type as neighbours.

Now, as an analogy if you look at a metropolitan city for example, where there are people from different states of the country everyone tends to or rather prefers to stay in an area where there are more people from their own state in a similar fashion the people in our model wish to have at least some other people of their own type as neighbours. So, what we will do is we will assume a common threshold t for each person if a person discovers that it is surrounded by fewer than t people of its own type he tends to move to a new cell such a version is called unsatisfied with his current location.

(Refer Slide Time: 03:48)



As an example if we set the threshold t to be 3 for instance and if you look at this node a labelled one it has 8 neighbours as you can see and out of these 8 neighbours 4 neighbours are of its own type and the rest 4 neighbours are of different type types. So, since the threshold is 3 it has more than 3 neighbours of its own type. So, we will call this node one to be a satisfied node if you look at the other scenario.

For example if you look at this node labelled one it is of green type and it has 8 neighbours out of these 8 neighbours 4 neighbours are of the different type and 4 neighbours are of the same type as the neighbour as the node one.

Now, since as an example if we set the threshold t to be equal to 3 let us say and we look at the node which is labelled one here as you can see there are 8 neighbours and out of the 8 neighbours 4 neighbours are of different type and 4 neighbours are of the same time as the given node one since the threshold t = 3. And this node one has more than 3 neighbours of its own time we will call this node one to be a satisfied node, right and if you take one more example and if you look at this node which is labelled 2 here it again has 8 neighbours out of these 8 neighbours 6 neighbours are of different types and 2 neighbours are of the same type as the node 2 and threshold t we have said to be 3 since this node 2 has lesser than t neighbours of its own type this node 2 will be called an unsatisfied node.

Now, as I already explained the unsatisfied nodes tend to move to new locations which are already empty and where they are likely to be to be satisfied their different versions of Schelling model in this context. For example, in some of the versions does the unsatisfied nodes we will move to those locations where of where there are more likely to become satisfied or in other versions the unsatisfied nodes will move to any random location. So, so there are different versions which you can implement.

Ah another thing is that whenever a node moves from one cell to the other cell it affects other neighbouring nodes as well for example, in this process some other nodes might become satisfied some other nodes might become unsatisfied. So, it basically changes the scenario of the cell of the grid, right. So, in every iteration, we choose some unsatisfied nodes and we move them to new locations and the cell and the grid structure changes and this keeps on repeating and we can analyze the structure that we get to in the end that is after certain number of iterations.

So, this is what we are going to implement we are going to implement the version of Schelling model where the unsatisfied nodes move to the random locations and we are going to run several iterations of this process we are we are basically going to recalculate the unsatisfied nodes in every iteration because that is; obviously, required because one shifting affects the rest of the nodes as well. So, let us get started with implementation now.