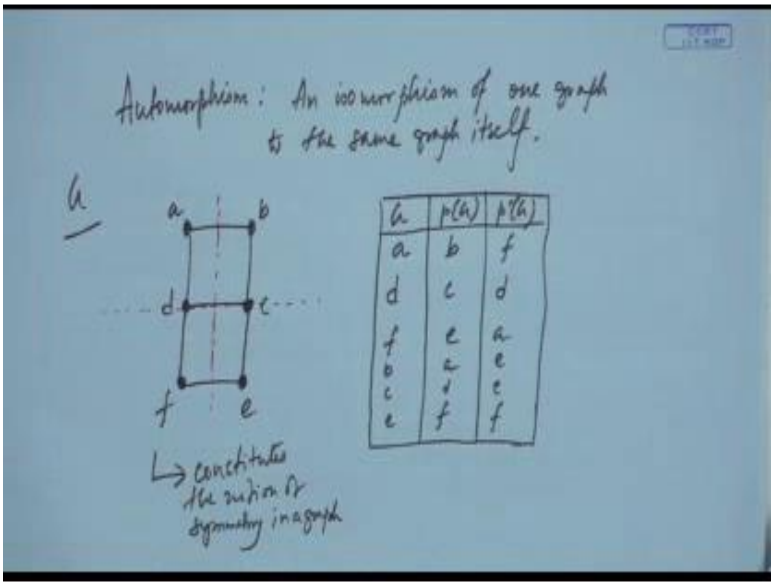
**Complex Network: Theory and Application**
**Prof. Animesh Mukherjee**
**Department of Computer Science and Engineering**
**Indian Institute of Technology, Kanpur**

**Lecture – 11**
**Social Network Principles – IV**

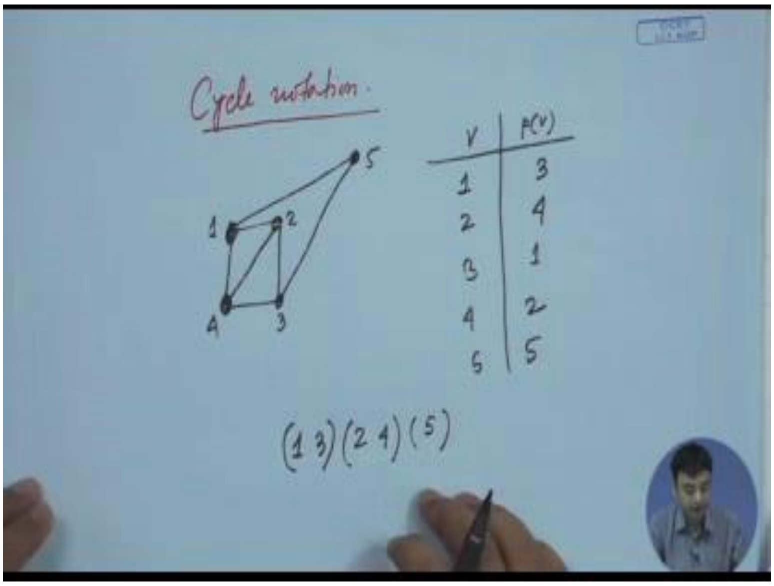
In this last lecture we have discussed how to quantify structural equivalence in principle and also we have got introduced to the idea of automorphic equivalence. So, if you remember.

(Refer Slide Time: 00:34)




So, we discussed that automorphism actually establishes the notion of symmetry in a graph, and it is basically a form of isomorphism of the graph to itself, and we also saw an example of this small square graph, and we saw that automorphism can be established along 2 lines of symmetry the, vertical axis as well as the horizontal axis. Now automorphism can also be represented using something called a cycle notation.

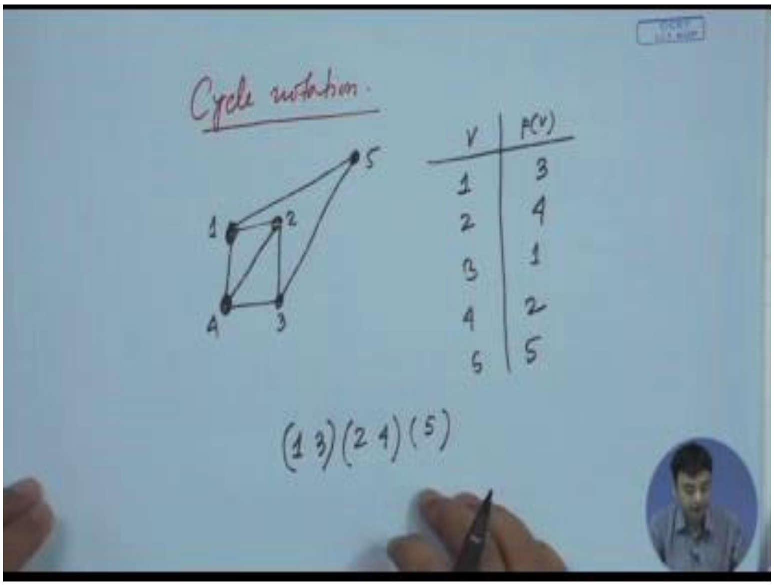
(Refer Slide Time: 01:05)




So, this is nothing, but a nice way a compact way of representing automorphism, for instance let us say we have this example graph here. So, you can immediately see that a automorphism of the following form can be established here in this graph; 1 maps to 3, 2 maps to 4, 3 maps to 1, 4 maps 2 and 5 maps to 5 itself.

So, basically since 1 maps to 3 and 3 maps to 1, we put them in a parenthesis like this. Also 2 maps to 4 and 4 maps 2, we put them in a separate parenthesis like this and 5 maps to itself. So, this is how you can represent a automorphism in a cycle notation. We can have another example; let us take this particular graph here.

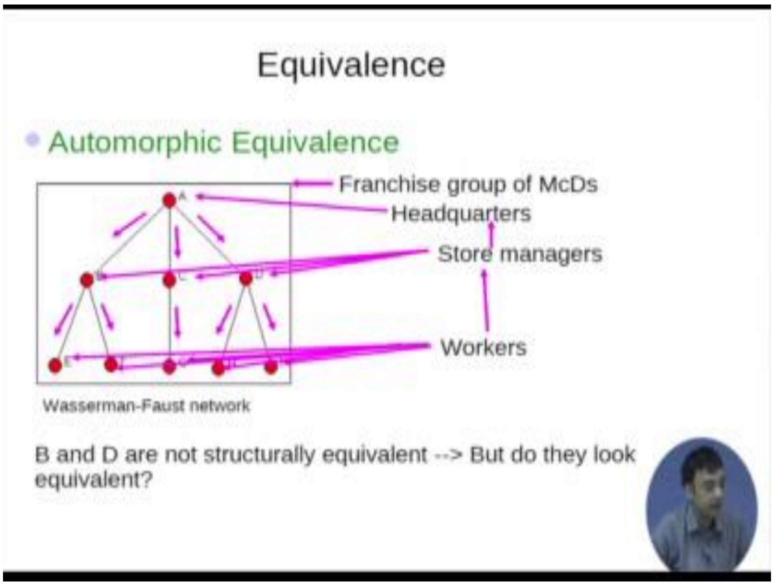
(Refer Slide Time: 02:30)




Again, we can establish the following automorphism a maps to b, b maps to d, c maps to c itself, d maps to a. So, then you have a cycle like a, b, d and c separately. So, this is a way, this is a very compact of representing automorphism in a graph.

Now, with this idea we go back to the Wasserman Faust example.

(Refer Slide Time: 03:25)



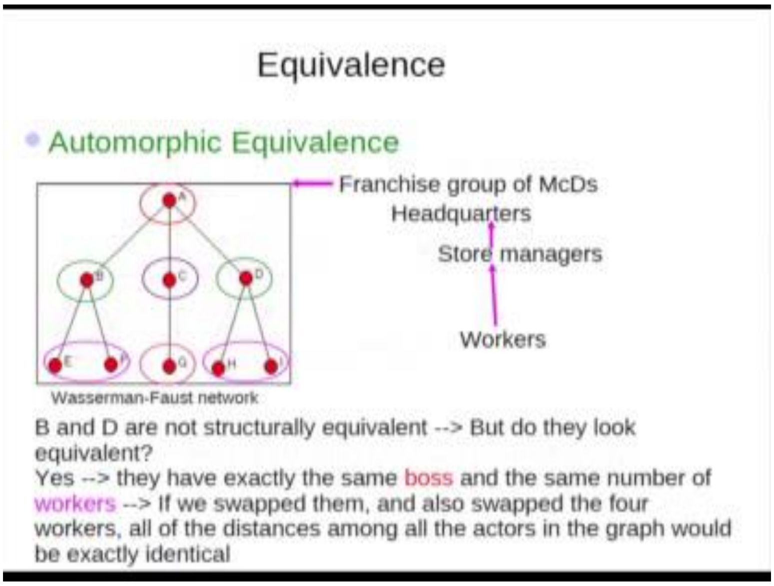

So, now look at the slides. So, for this Wasserman Faust graph again we can define from the same graph we can define an Automorphic Equivalence. So, imagine. So, for this let us have an, let us motivate our self using an example. Imagine that this box rectangular box here represent the franchise of Mc Donald's. So, this is 1 of the well known (Refer Time: 03:50) if you are interested you can also replace it by KFC. Now, at the bottom of the graph, let us assume that these are workers in the franchise.

Now at the middle level say we have the store managers, and at the top level is headquarter. So, the ruling directions are given by the arrows, the reporting direction will be just the opposite arrows. So, the arrows that you see here are the ruling directions, where as the opposite direction will be the reporting direction. Now the point is if you look at this particular example, you can immediately observe that although b and d are not structurally equivalent, they are not structurally equivalent because they do not share the same set of papers, but then they somehow seem to be having a similar role.

See both of them have actually, same set of workers working under them same number of workers I am sorry not the same set, but the same number of workers working under them, and they report to the same number of headquarters. So, these 2 people, these 2 store managers have a very similar sort of a role. And they are a little different from c, because c manages only 1 worker. Whereas b and d actually manages two workers, probably they have the same size shops they manage 2 workers and they report to 1 person, to one headquarter.

Now, how to quantify basically the equivalence between b and d, this is done by the notion of automorphic equivalence, which we have studied; you see that b and d are actually defining the symmetry in this graph if you swap b and d along with all its neighbors, then you do not see any observable change. So, you can swap it perfectly and that actually establish symmetry. So, that is why b and d can be thought of as symmetric to each other because they establish automorphic equivalence.

(Refer Slide Time: 06:08)

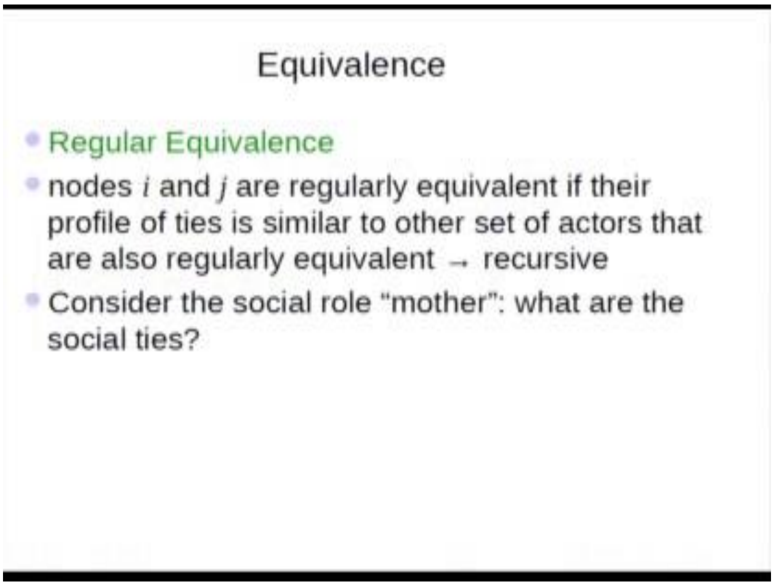



So, that is all about automorphic equivalence. So, given this idea of automorphic equivalence, now we can basically define the automorphic equivalence classes. Now if pair of nodes is structurally equivalent, they are definitely going to be automorphic equivalence. Because they share the same set of neighbors exactly the same set of neighbors.

So, structurally equivalent nodes are always automorphically equivalent, but not vice-versa so; that means, e and f are automorphic equivalent, similarly h and i are automorphic equivalence because they are already structurally equivalent, g is a separate class in itself, c is a separate class in itself, but now b and d form in the fall in the same class that is why I have colored them with the same color. d and b actually form a similar class, a same the same automorphic equivalence class and a is a separate class in itself.

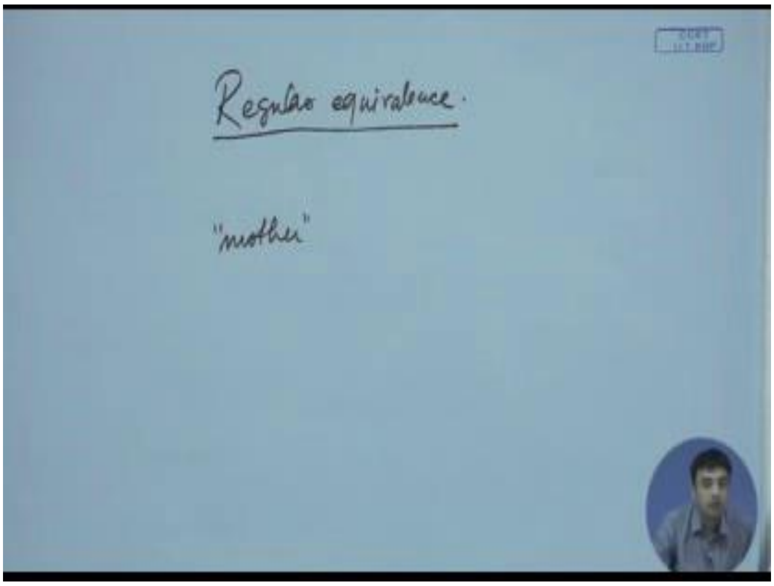
So, this is where you relax the definition a little bit from the structural equivalence to automorphic equivalence by translating exact substitutability to symmetry. So, that is how you translate or relax the definition of structurally equivalence, to the next i d of automorphic equivalence which becomes handy in this sort of examples of Franchise, Mc Donald Franchise or similar such organizational structures.

(Refer Slide Time: 07:34)




So, now there could be a third type of relaxation or a third type of equivalence, which is usually called as regular equivalence.

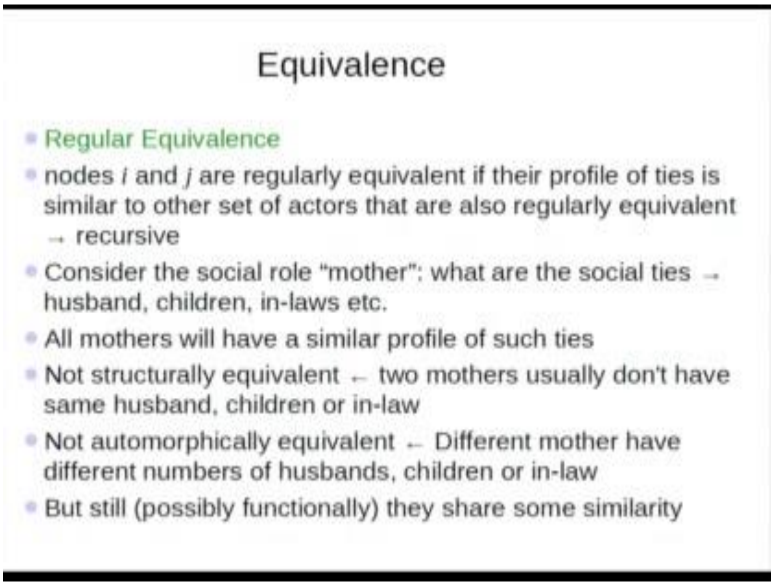
(Refer Slide Time: 07:43)




So, the idea is that. So, if there are nodes in the network, which have similar profile of ties to other nodes that are regularly equivalent among this. So, this idea is very much recursive. So, this idea is very similar to the idea of recursive definition of page run that we have got introduced earlier. So, basically what we say is that 2 people are, if they have the same profile of ties to other people in the network who are already known to be regularly equivalent, then these 2 people are also regularly equivalent.

So, if 2 nodes x and y have a similar profile of ties to other nodes in the network, which are already known to be regularly equivalent; then x and y are also known to be regularly equivalent; for instance take the example of “ mother” the relationship mother. So, if you consider this relationship. Now, “mother” each mother in the social network.

(Refer Slide Time: 09:59)



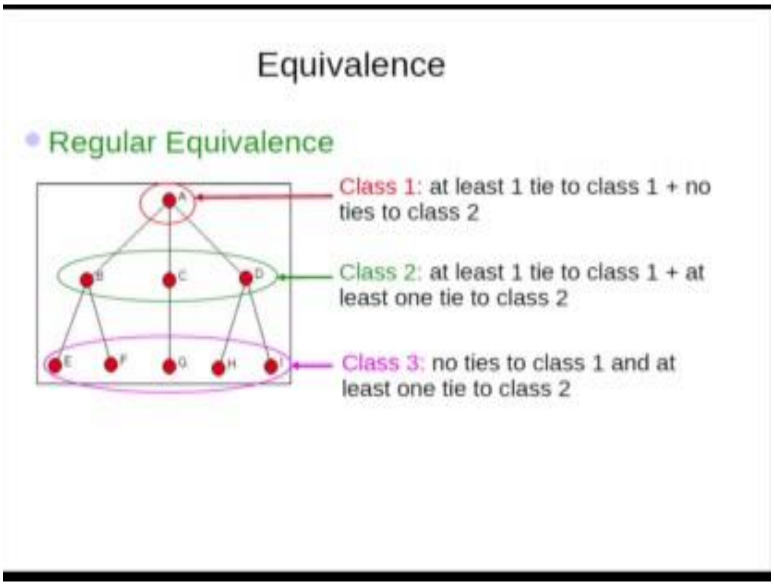

If you look at the slides I write it clearly, I demonstrate it clearly there. So, all mothers will have a similar profile of ties, they will have husband, a set of children a set in law, every mother in this social network will have a husband, will have a set of children and a set of in laws. And every one of them will have a similar sort of a profile like this, but then none of the two mothers are structurally equivalent.

Because two mothers usually do not have the same husband, or the same children. So, that is not possible. So, 2 mothers are structurally equivalent, also two mothers are not automorphically equivalent because, two mothers might have different numbers of in laws and different numbers of children. So, they it is not mandatory that two mothers should have the same number of in laws and the same number of children.

So, neither they are structurally equivalent nor they are automorphically equivalent. So, they, but then the notion of motherhood actually define the similarity between them. And these similarities are actually captured through regular equivalence as we say. So, the idea is that, they are these; these actors are more functionally similar. So, they perform similar functions as mothers. So, regular equivalence is actually one of the measures which try to capture functional equivalences between pairs of nodes in a social network.

Remember that they are very, very different from structural equivalence nodes; because two regularly equivalent nodes like two mothers might not have, might not share the same will actually never share the same husband or the same set of children. And they might not have the same number of in laws or same number of children to the automorphically equivalent. But still they are there is a notion of similarity between them, and that is what is we quantify using regular equivalence and we call that the basis of this particular equivalence is some sort of functional similarity.

(Refer Slide Time: 11:13)

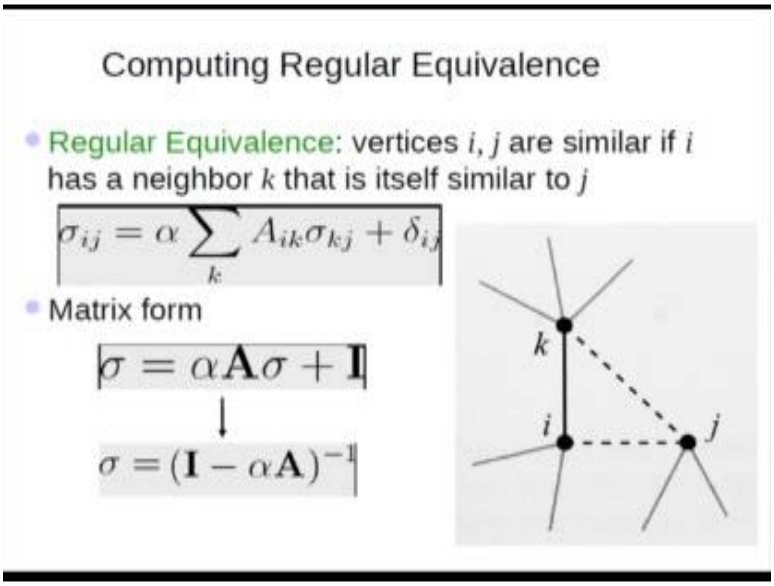



So, again taking the Wasserman Fauts graph you can define such a nice regular equivalence. So, for instance let us look at the three classes here. So, here regular equivalence will be divided or segregated into three classes; the class 1, the class 2 and the class 3. See class 1 node a; it is a class in its self, it is a regular equivalent class itself by virtue of having at least 1 type to class 1 members and zero ties to class one. So, node a is 1 class the class 1. Now this node this class members of this class will have at least 1 tie to the members of class 2, and 0 ties to the members of class 3.

On the other hand class 2 will have at least 1 tie to members of class 1 and at least 1 tie to members of class 3. On the other hand class 3 will have 0 ties to members of class 1, and at least 1 tie to members of 2. So, in this way the 3 classes of regular equivalences are developed in this particular Wasserman Fauts example. In this example, what we see is that by virtue of its profile of relationships with the other classes that are regular equivalence class is defined. So, for instance the first class; class one will have actually 1 tie to the second class, that is class 2 and no ties to class 3.

Similarly, class 2 members will have 1 tie to class 1 members and at least 1 tie to class 3 members, and class 3 members will have 0 ties to class 1 member, and at least 1 tie to class 2 members. So, in this way we define the concept of Regular Equivalence. Now this can again be quantified as we did quantification for the page run definition. So, regular equivalence for instance again take the example of this small this graph here in the slides.

(Refer Slide Time: 13:24)




So, suppose we are wishing to find out the regular equivalence between a pair of node i and j. Now we know that, there is an edge between i and k and k and j are known to be regularly equivalent. k is a neighbor of i and k is known to be regularly equivalent to j. So, we define the regular equivalence between i and j in terms of the knowledge of its regular equivalence with; with in terms of the knowledge of the regular equivalence of j with k and that i is a neighbor of k.

So, we denote this as sigma i j, that is equal to nothing, but alpha which is a constant which is a constant pre-factor into A i k sigma k j. So, sigma k j is basically the expressing the regular equivalence between the extent of regular equivalence between k and j plus a small constant delta i j. So, we are assuming that there is some small factor delta i j. That chronicle factor between i and j, basically we are assuming that in this case delta i j for all j's except i will be 0.

So, we will assume that delta i-i is equal to 1. So, basically you can be regularly equivalent to yourself that is the idea. So, to start with you are regularly equivalent to yourself only, and then you try to find out evidences of other regularly equivalent classes from your neighborhood and try to establish regular equivalence with them. So, that is that actually gets absorbed into this formula.

So, you have alpha sigma of k a i k which is that is a i k is indicating whether, i is a neighbor of k. So, you are looking at the adjacency matrix and see whether a i k is equal to 1 that is i is a neighbor of k into sigma k j, which is the extent of regular equivalence between k and j that is hope to be known, plus the initial similarity between the node and itself between the regular initial regular equivalence between the node and itself that is expressed by delta i j.

So, delta i j is actually zero for i naught equal to j, and equal to 1 for i equal to j. So, you start with a prescription that each node is regularly equivalent at least itself, from there you have sigma equals to say if you try to quantify this, you have sigma equals to alpha a sigma. So, just writing it is in the matrix format just as we wrote in case of our page run vectors. Sigma is equal to alpha a sigma plus i which is the delta i j. So, as I said is the identity matrix you have it has diagonal entries. So, I basically is the identity matrix, then it will only have the diagonal entries, whereby we assume that the nodes are only regularly equivalent to themselves to start with.

Now, with the evidence from the regular equivalence of their neighbors with other nodes they will have a, they will establish a regular equivalence with them. So, from there you can calculate the regular equivalence sigma equal to i minus lambda a inverse. So, this is exactly as we computed our page rank formula. So, we stop here.

So, we have actually to summarize we have actually discussed the roles, that different individuals play in a network and their relationship to other nodes in the network, we have quantified this roles in at least 3 different forms, the structural equivalence, the automorphic equivalence and the regular equivalence. We have seen in each case how one can try and quantify each of these ideas.

Next day we will start off with community structure analysis.

Thank you.