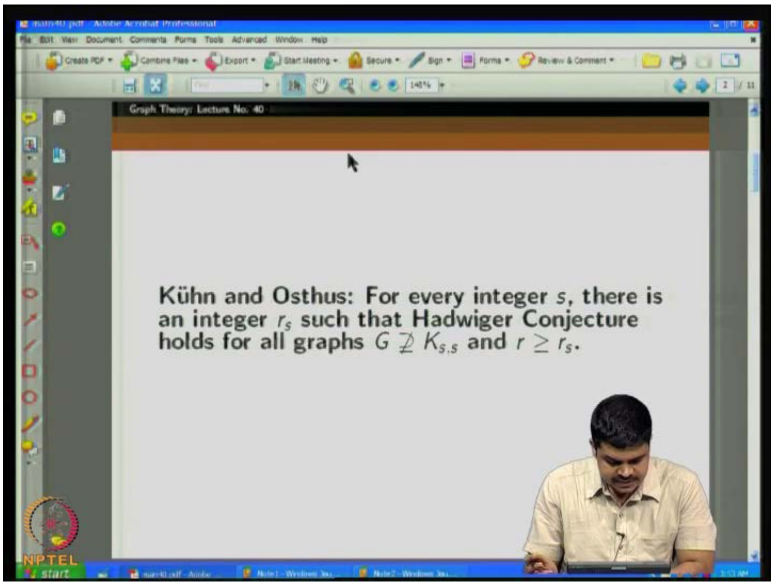
**Graph Theory**
**Prof. L. Sunil Chandran**
**Computer Science and Automation**
**Indian Institute of Science, Bangalore**

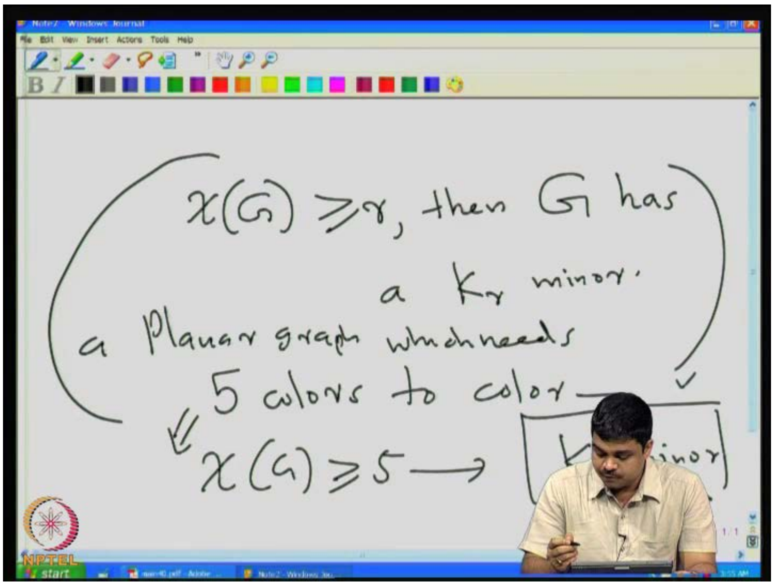
**Lecture No. # 40**
**More on Graph Minors, Tree decompositions**

(Refer Slide Time: 00:28)




Welcome to the 40 th lecture of graph theory. This is the last lecture. So, in the last class, we looked at Hadwiger conjecture to remind you, what was Hadwiger conjecture? So…

(Refer Slide Time: 00:53)




So, the Hadwiger conjecture stated that if the chromatic number of the graph greater than are equal to r, then to graph G has a k r minor in it. What is a k r minor? A minor on minor that size on morphic 2, the complete graph on r vertices, so for example, we show that the if the planar graph requires for example to understand this thing, if the planar graph. If suppose there is a planar graph which requires 5 colors to color, a planar graph which needs 5 colors to color that means the chromatic number of G being greater than equal to 5, then we will have a k 5 minor in it. But then we know that the planar graph cannot have a k 5 minor; why, because if you contract the edges of a planar graph it will still be planar, and then this minor operations will always the any minor of m planar graph has to be still planar, but k 5 is not planar.

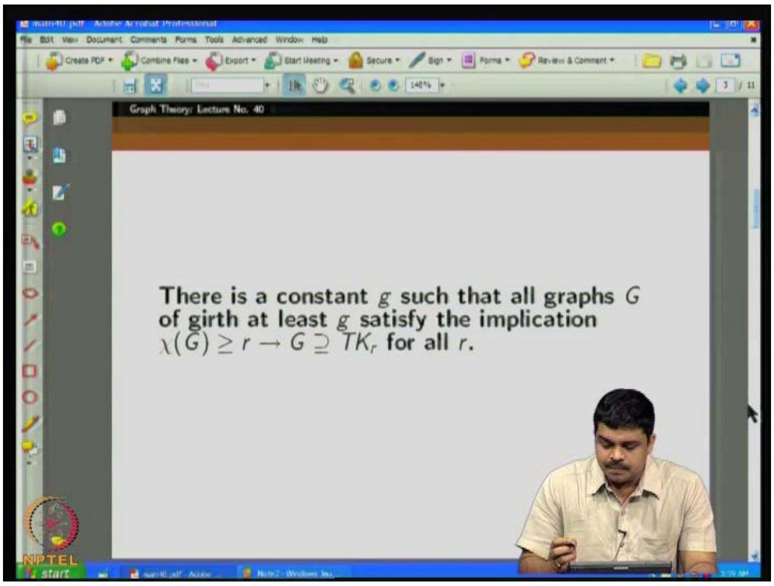
So, this Hadwiger conjecture if true will imply the 4 color theorem. So that means every planar graph has to be colorable using 4 colors. So, of case this is the much stronger statement, much more general statement then the 4 color theorem. So, this size that it is not only for planar graphs, every time a graph can be colored sorry, if a graph chromate number is at least r then there x is a k r minor. In other words, if a graph does not have a k r plus 1 minor we are sure that it can be vertex colored using at most r colors. This is what Hadwiger conjecture size.

Now, the Hadwiger conjecture is not a proofed, it it remains is one of the most challenging difficult open problems in graph theory. But several special cases are already proofed like for in sense when the chromatic number is equal to 5 that correspond to the 4 color theorem and little more than that in sense that is why the 4 color theorem follows from r equal to 5 case, so r equal to 6 case is also proofed the robots and (( )). And then but let a cases are still open, well then of case there are very several special clauses of graphs for which Hadwiger's conjecture is true, For examples, we can considered the perfect graphs where the Hadwiger's conjecture is trivially true. Why is a trivially true, because a perfect graph if you remember the definition of the perfect graphs, they are the graphs whose chromatic number is equal to clique number not only for the graph also for all the induce sub graph. That is in material for us in with respect to this problem.

But we know that for perfect graph the chromatic number is equal to clique number, so the there there x is a clique minor if the chromatic number is equal to r, there x is a clique itself not just a clique minor, there x is a induce clique there with r vertices. So, there is a clique minor also. So, the perfect graphs it is trivial. So, now the question is other clauses of graphs. So, for line graphs it is non to be not we conjunct this is not to be true. For proper circular or graphs, it is non to be true. So, there are several classes of graph so which the Hadwiger conjecture is non to be true.

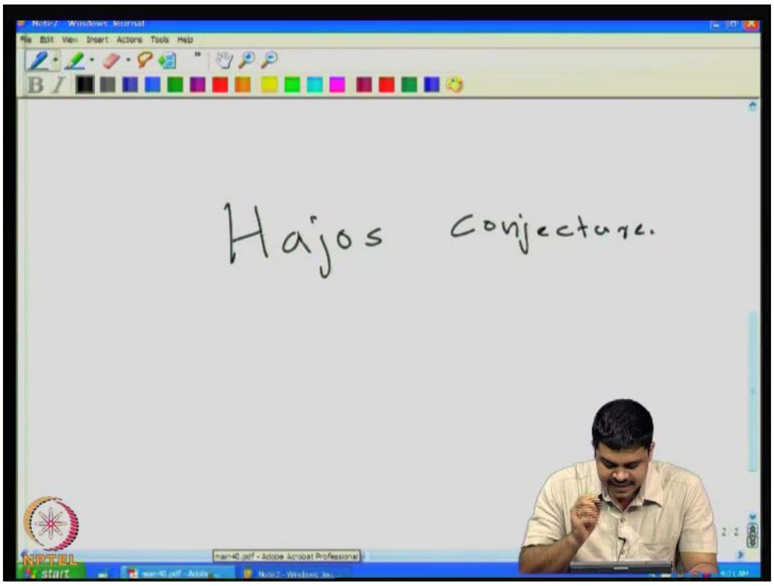
So, in this context, so we carry on from this point, and we also want to mention some of this special cases which was proofed by Kuhn and Osthus. So, suppose s is an integer and then there x is an integer r s a correspond as a function of s you can find another integer r s. Such that the Hadwiger conjecture holds for all graphs which does not contain k s s as a sub graph in it, k s s as a sub graph means that complete bipartite graph with one side s and other side s, if it is not part of the graph then we can say that the Hadwiger conjecture is true; not for every value of r then r is sufficiently large that is what it says. If so begin a values of r the Hadwiger conjecture will get true is what this result of Kuhn and Osthus.

(Refer Slide Time: 06:28)




And now, let us come to the topological minors up to Hadwiger conjecture concerns with the existence of a complete graph as a minor. So, what about complete graph is a topological minor? So, is it true that if the chromatic number is greater than equal to r, we have the complete graph as a topological minor itself, of case to asking a complete graph as a topological minor is a much stricter stronger requirement, then asking for a complete graph as a minor. So, it is less possible less probable, but this whose also conjecture earlier. So, like Hardwiger's conjecture there was also a parallel conjecture called Hajos conjecture.

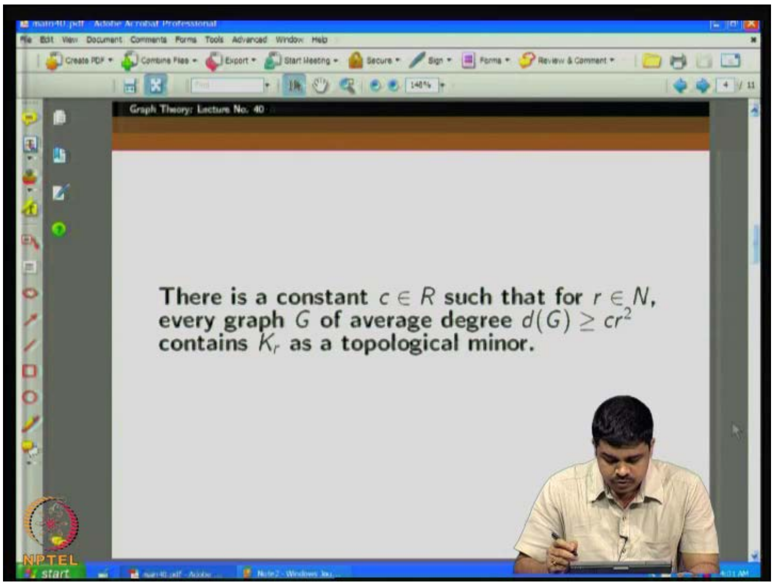
(Refer Slide Time: 07:33)



Hajos conjecture which stated exactly this, if the chromatic number of a graph is greater than equal to r, then there x is a k r topological minor for the graph, but it turned out that the conjecture is strong after quite some a number of I mean after several years; it was proof that there x is a counter example for Hajos conjecture, it is not true in general. (( )) after that it was even shown that not just one counter example, for almost all graphs to Hajos conjecture is wrong. So, in a probable (( )) you can randomly select a graph and show that its chromatic number is something, but it will not have a k r minor of that corresponding chromatic number. And of case several other methods to construct counter examples for Hajos conjecture is known now.

So, Hajos conjecture turns out to be wrong, but still it was quite popular for several years. It was also very well studied, so therefore several results regarding in the direction that is also there. So, for instance there is you can compare this result of Kuhn and Osthus with a previous result. They say that if the girth is a little high then the Hajos conjecture is indeed true; Hajos conjecture may be false, but only when the girth is small. So, the they say that there exists a constant G such that all graph G of girth at least G satisfy the implication. If the chromatic number is greater than equal to r then it has a k r topological, k r has a topological minor for all r.

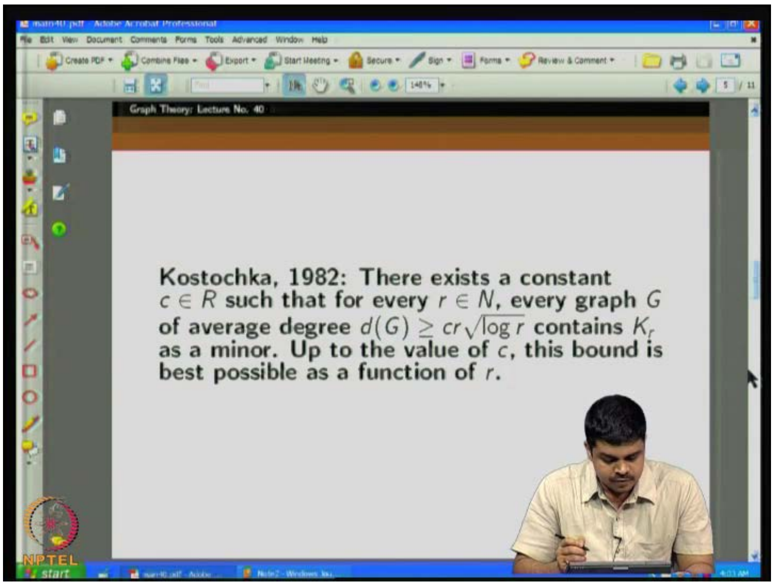
(Refer Slide Time: 09:21)




So, the another way of asking another line of question is questioning is so the the the, so what should be the average degree in terms of r such that if I know that the graph has average degree at least this much, then we are sure that there exist a k r topological minor in the graph. It was shown by madder that the if the average degree is greater than equal to some constant time r square then we do have a k r as a topological, I have k r as a topological minor in the graph. We need the average degree to be the it to be quadratic in r r square c times r square. And of case, more are less this is the only possible thing, we can show that there exists some cases when it is below that you cannot expect a k r topological minor. In general that solve we can tell about in term the average degree requirement is that.

But suppose we do not want a topological minor, we only want a minor - k r minor, then what should be the average degree requirement; how much should be the average degree, so that we are sure that if the average degree is that much then there is a k r minor for the graph.

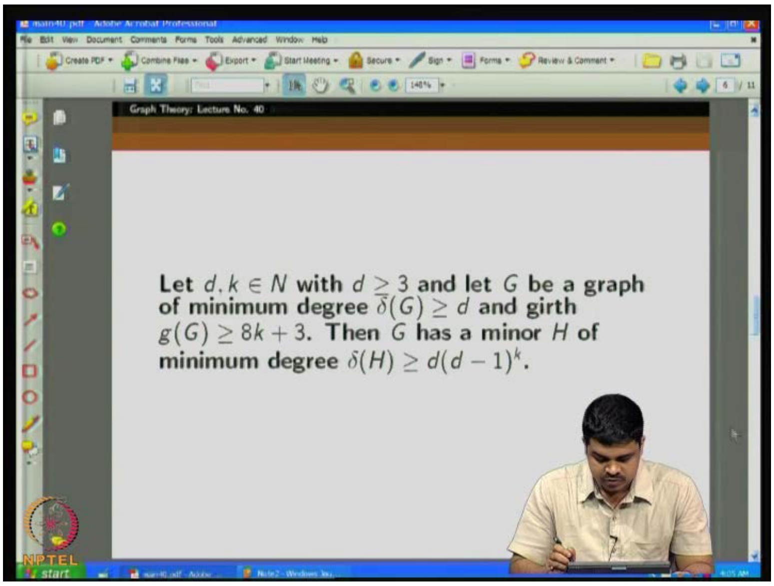
(Refer Slide Time: 10:59)




So, this question was answered by Kostochka. So, he showed that there exists a constant such that for every integer r every graph G of average degree at least c times r root log r, r root log r times constant contains k r as a minor. We can compare it with the previous result which says that if it is the average degree is at least c r square, then it contains k r as a topological minor; here the requirement is much less, we do not need r square, we need only r root log r. So, and it is also known that up to the value of c, this constant can be improved a little bit probably then what Kostochka is an I it looks like it is also known which is the best constant. This bound is best possible as a function of r. We cannot hope to improve much; so, this is the best possible.

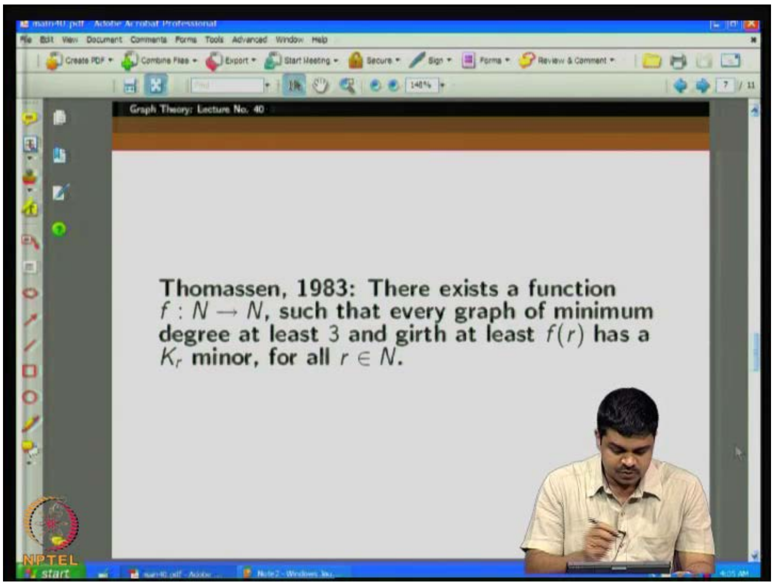
Now, we are not giving the proofs of this result neither the proof of the earlier result, because here all more complicated. It will to bet get traits of this kind of results, we will look at one result the proof by proof of a which is given by distal.

(Refer Slide Time: 12:38)




So, it is show that if the girth of the graph is high then we can expect some dense minors in the graph. So, with that is about the of case, so we we will look at this result.

(Refer slide Time: 12:41)

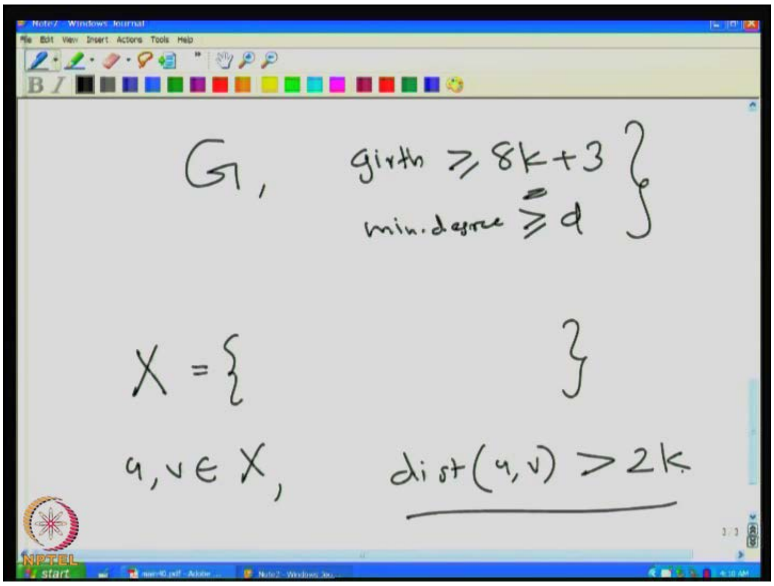


This first it was proved by Thomassen, he showed that see of case, the earlier result says that if you have the average degree is at least c r root log r, then we do have a k r minor. But what if the average degree is not that much? Average degree… So, suppose we just showed that minimum degree is 3, whatever is r we are assure that the minimum degree is 3 there is no more assurance on the average degree in terms of r we are not told that the average degree is more than some function of r. But we know that every degrees at least 3.

Then Thomassen found out that if we simply increase the girth, if the girth is greater than equal to some function f(r), here girth is a function of r then we do get a k r minor. This function is to be found out which is the correct function. So, but he is found out the there exists a function that if the girth is begin up that ways greater than equal to that f of r, then even with minimum degree 3 without any other restriction an average degree, we can expect to get, we can get a k r minor. This is interesting, because when we fix the minimum degree at 3 an increase the girth without increasing the minimum degree same. Then the graph will become spars looking from one vertex, it will look like tree like structure is going to some distance right, it will become more and more spars. But still we get as this girth requirement is increase, we do get bigger and bigger clique minors; we are assure that a k r minor exists if the girth is greater than equal to f of r right.

So, now we will do a proof of this result, but before that we need a lemma develop by distal so so that we can estimate this f of r also. So, this is the lemma, let d comma k element of n that means d and k are two integers with d greater than equal to 3. And let G be a graph of minimum degree delta greater than equal to d, and girth greater than equal to 8k plus 3, then G has a minor H of minimum degree, delta of H greater than equal to d times d minus 1 raise to k. So, so what this result says is, if your girth is high, so this value given here girth is greater than equal to 8k plus 3. And also the degree is high, here we are not saying that the degree minimum degree is only low at least 3 we are saying that minimum degree is at least d, then we say that in terms of d and this k k is in fact G minus the girth minus 3 by 8 right. We can get a lower bound for the minimum degree of the minor that means we will get dense minor that is what it is says - high minimum degree minor H is available right. So, how do we prove this thing?

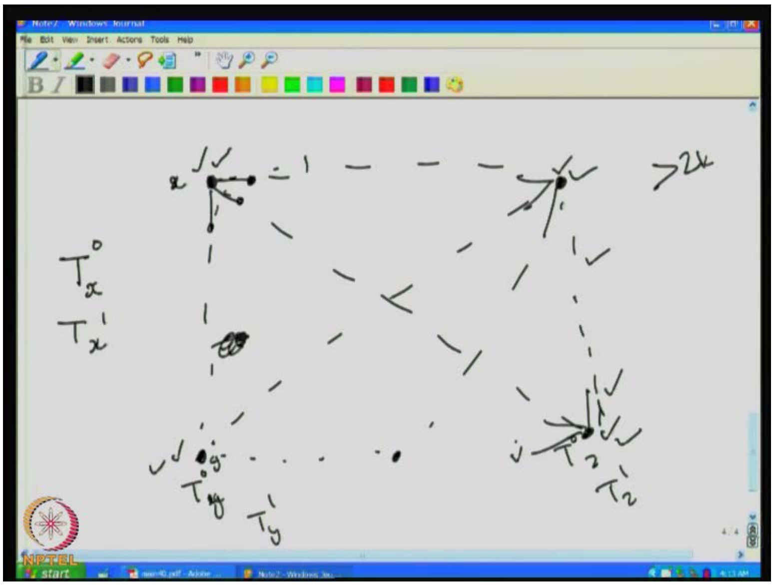
(Refer Slide Time: 16:25)




So, the to prove this thing let us consider the the, so a graph G with the given properties that means its girth G is greater than equal to 8k plus 3 this is what we have to remember. It is a minimum degree is greater than equal to d. This is what we have to remember. Now, we will collect vertices from G that is a x. So, this are the set of vertices from G such that between any pair of vertices in G, the distance is for instance if u comma v element of x then we need distance between u and v - shortest distance between u and v strictly it greater than 2k. It is k being this k right strictly greater than 2k. So, we can start with one vertex, if we can add one more, we add it right; so, we if we can add one more to this we will add. So, it was to the extend that the once you get to the situation that we cannot add anymore vertex to this. So, one of the already existing vertex will will have a problem if you add one more vertex to it, any vertex any outside vertex is added to x.

The distance between the one of the already exists at least one of the already existing vertex will become less than equal to 2k that is why we cannot add. So, such in such a situation in such a situation, so we stop we say that it is a maximal such set with this property that between pair wise distance between the vertices include (( )) is greater than 2k.

(Refer Slide Time: 18:13)

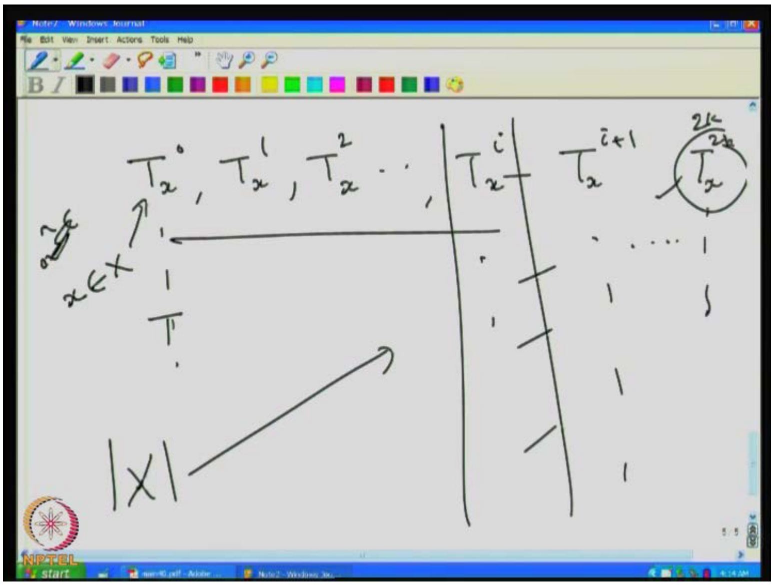


So, now what we going to do is so let say this are the vertices of x, some vertices though some how I am just so the thing is any distance here, a shortest distance here, is not the geometric distance, it is the distance in the graph with the shortest distance in the graph; any pair wise distance will be greater than 2k. Now, from each of these vertices we will develop a tree, first what we have doing is we will simultaneously develop a tree. So, for x if this the vertex x then we will call T x first we construct T x 0, and here this is y we will construct T x T y 0, this is said that is sets 0 and so on. And then after that in the next step we will construct T x 1 here by adding a few more vertices, then we will here we will construct this T y 1, and here T z 1 and so on.

So, what is this T x 0? T x 0 is this single vertex itself right, each of them will be just single vertices. Here this is x only, here this is y only, this is, and T x 1 is what? We will look at any vertex we will collect the vertices which are at a distance of exactly 1 from this thing, the reachable at a distance of say from the not already taken vertices, we will take a vertices which are reachable in 1 step from any of this vertices. So, then we will add to one of them for instance like this. So, in principle for instance if there is a vertex like this which can be added to this and this will add to one, only one of them right, the arbitrarily choose to which…

So, then we will get a few trees here right, and then in the suppose in the i th step, so in the next step what will we do? We will take a all the vertices which are reachable in 2 hops, it a distance of 2 from each of this vertices in any of this vertices, then they will be attach to one of the trees which are already developed. It should be possible, because any way if you track shortest path to the set x, the the just previous vertex is already attach to some of this trees and then we can attached it to attached to that right tree.

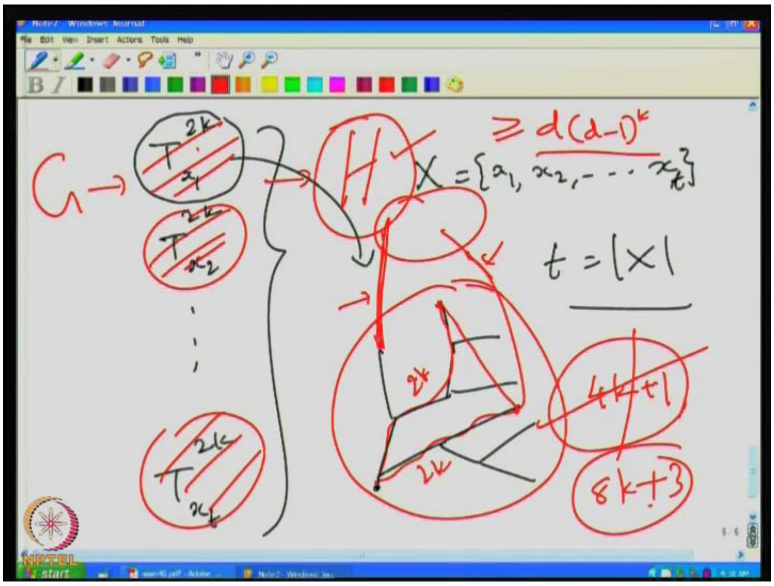
(Refer Slide Time: 20:41)



So, like that we can construct T x 0, T x 1 like that we can keep constructing. So, T x 2 so up to T x i, and this will be done for each vertex in x, say for each x element of x we will each x element of x we will construct this kind of trees right. So, in the i th stage we will have several such the one tree corresponding to each of those vertices in x, the this many trees will be there right. Now, to construct the i plus 1 th tree say T x i plus 1 this set of trees, what we do is we consider the vertices which are not already taken in fact, which are at a distance of exactly i plus 1 which are reachable in i plus 1 hops from the set x.

And then each other's vertices will be attached exactly 1 trees in it 1 tree in it. It is not that there is only one shows, it is possible that some vertices can be attached more than one trees, but we will select one of them and then add to that. So that the trees remain disjoint. And it is clear that when we reach T that 2k th step - the T x 2k and other things for, so we have already covered all the vertices of the graph that means every vertex of the graph belongs to one of this trees. So that is become a partition of the tree.

(Refer Slide Time: 22:09)



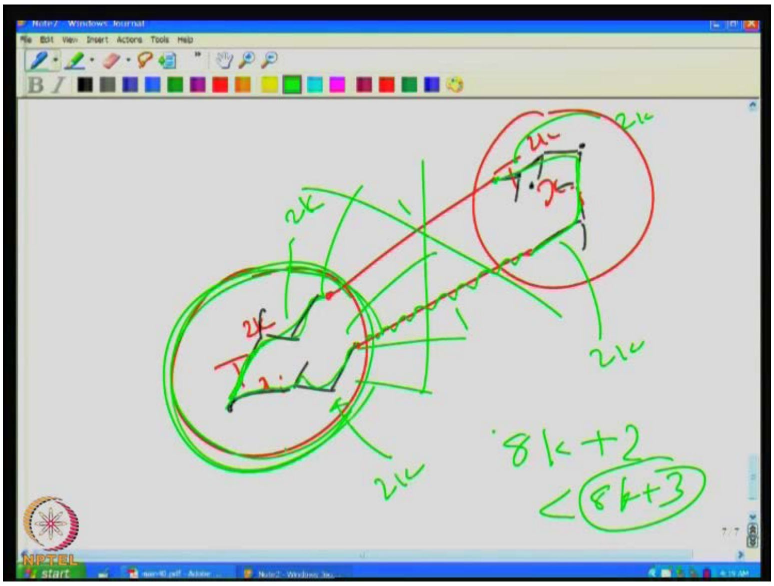
Now, the the some of this properties of this trees for instance I can call it T x 1, so let say this set is X equal to x 1, x 2, x something, x t or something, so then we can say x 1 2k, x 2 x 2 2k this are the trees we have, T x t 2k where we have t is equal to the cardinality of X right. These trees are certain properties. For one thing each of them are induced trees in the graph. Why are the induce trees? So, let see, suppose it is not, so there is a tree here, so this is one of the trees which we created; suppose there is some edge here, some edge here, but then I know that this vertex the distance from here to here is at most 2k, and the distance from here to here is at most 2k. So, 2k plus 2k 4k plus this edge together will make 4k plus 1 cycle, but we know that the girth of the graph is 8k plus 3 are more.

So, this is definitely less then this, so this is not possible such a cycle will not exist. So that is why this k(s) has to be induced, we cannot have an extra edge there; we cannot have an extra edge, because if you have an extra edge we will get a cycle, and this cycle is definitely of length at most 4k plus 1. So therefore, these are all induce k(s). That means any edge which is going out of this tree has to go to other trees that connect it to other trees.

So, plan is to contract each of this induced trees, each of this induce trees and get a minor. So, let us call it H. So, this will be the minor which will be created H, which the minor H which will be created from the this graph. So, our intention is to show that this H minor is the kind of minor we want; that means of minimum degree greater than equal to d into d minus 1 raise to k. That is what we claimed in our proof right d into d minus 1 raise to k. This is what we want to show.

To show that first observation is all the edges going out of this trees, because this is an induce tree should go to other other trees other brand set, this are all brands sets now right. But is it possible that one edge going out of here, and one edge going out of, both goes to the same tree, because the problem is, if you if you contract this and this contract that will become parallel edges and then we will loose one edge unnecessarily. You do not want to loose any edge. So, we want to say that each of the edge which is going out of this thing this tree will go to different, different trees. So now, it will not go to the same tree, because if two different edges from this tree goes to this goes to the same other tree then when we contract this tree and contract that tree, it will become parallel edge and one of them we will lose. We want all the edges which are going out of this tree to become and finally, count as the degree of this thing, finally to contribute to the degree of this contracted vertex. So therefore, let us see (( )) analyze the case.

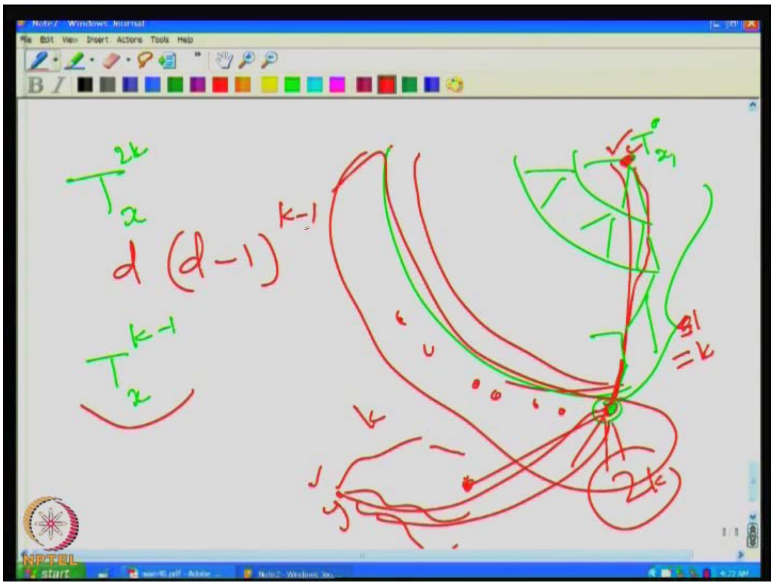
(Refer Slide Time: 25:31)




Suppose there is T x i 2k and there is this T x j 2k. We just possible to have two edges going like this, so if there are so the tree here may be like this right and is if there are two such two such trees, then the problem is, so what will happen to this cycle? So, this comes to the route, our root are wherever, the common answers for this thing and then come here. And then similarly, this will go to the common ancestor of these two vertices, and then come this tree. And this root this path is at most 2k, this path is also at most 2k, this is at most 2k and this is at most 2k, and this are here 1 and 1; total 8k plus 2 that is still less than 8k plus 3. So, because the girth is greater than 8k plus this cycle cannot exists. So, this will be, will not happen, this will not happen. That means from this tree you can only have one edge going to the other tree - any other tree.

So, (( )) in the contracted graph to know the degree of this vertex - the contracted vertex this vertex, we just need to count the number of edges going out of it. If we get a lower bound on the number of edges going out of it, we get a lower bound for the degree of the each vertex right. How do find the lower bound for the number of edges going out of it?

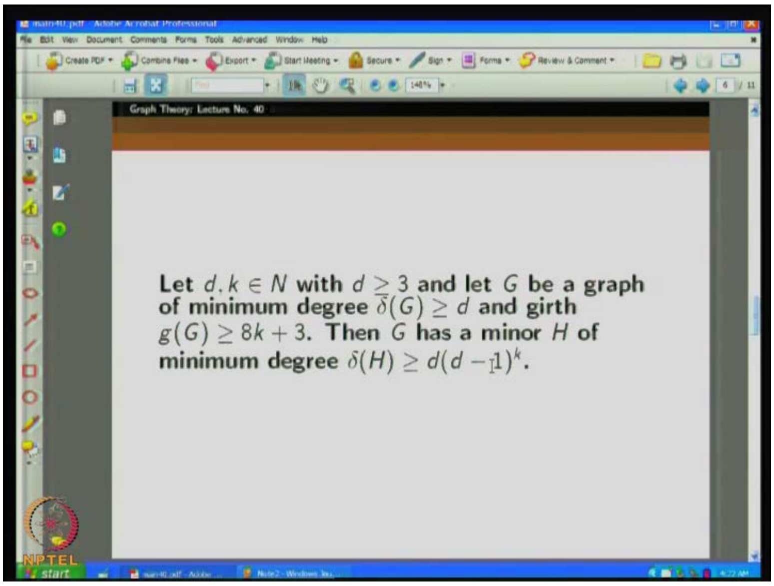
(Refer Slide Time: 27:20)



That is done like this, because for instance is T x is first T x 0 then T x 1 0 then this is here, we got T x 1 1 then we in the next stage we got T x 1 2, so up to the i th stage, suppose we say when we are consider in the k minus 1 th stage we had our tree finally is T x 2k. But consider this tree T x, the stage at which the k minus 1 step, how many of the neighbors will be captured? So, any vertex at the k x th stage, is it does it get two choices? That is the question. Suppose this got attach to this tree at this stage this got attach to this tree at this stage, but is it possible that is could have been attach to another one? So, the point is, it cannot be, because the distance from here to this tree is at most k; so exactly k that is why it is getting attach to this tree, but then if suppose there is another y to which y tree from starting from y to which it can be attached, then the distance from here to here also will be k, and the total distance from this point to this point will be k plus k equal to 2k. But that is we know that this distance has to be at greater than 2k. So that is not possible.

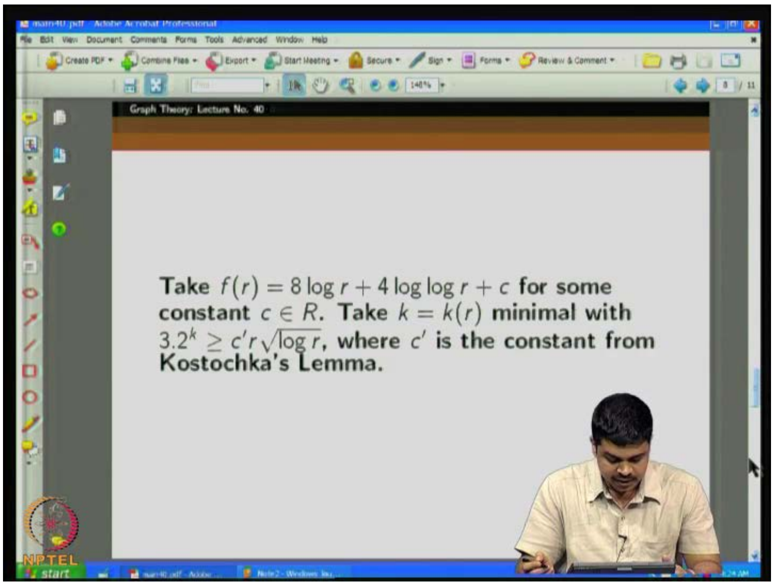
So, all the vertices at distance k from this x will be attached from this tree this tree only. That means at least in the k th when you moving from k minus 1 stage to k th stage, we will be capturing at least how many vertices, because here we have you know all the vertices up to here all the vertices will be. This the number of leaves of starting from here, if we had captured all the neighbors at every stage up to here that is like that only, because the first level we will get all the neighbors, then this level all the neighbors of neighbors will come, this level all the, because they do not they would not be claim by another tree, because there are too for away, all other trees are too away at this up to this k minus 1 level. So therefore, this number of vertices can be easily estimated, because that is at a distance of k that is definitely d into right so right d minus 1, because here, the first level we have d and they d minus 1 raise to k minus 1 right. So, the first level we have d, in the second level…

(Refer Slide Time: 30:15)



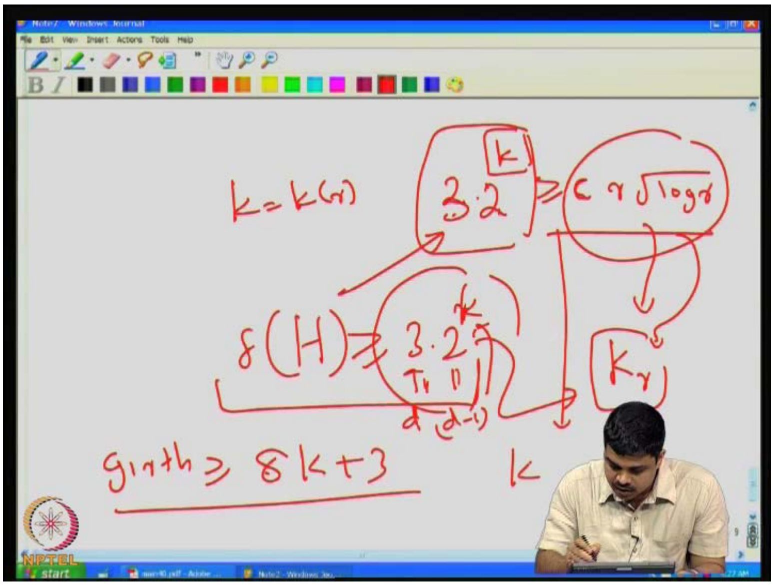

And then the number of edges we the, how many edges will go out of it? So that is very clearly the required number, this number will come d into d minus 1 raised to k. Because we had in multiply again by d minus 1, so this will give you the minimum degree - lower bound for the minimum degree of H, will get a minimum degree of the lower bound for the minimum degree of H. So, we do has a H minor for G with where the minimum degree of H is greater than d into d minus 1 raise to k. This k comes from the girth, because the girth is define in in terms of k 8k plus 3 right; d is the minimum degree of G itself. So that this we have a minor who is minimum degree is exponentially higher, I mean that exponent is coming from the girth, then the the minimum degree of original graph right. We somehow we concentrate at the get the the the we made the graph dense. And now we can apply Kostochka as a result on this thing, because there is a graph with high minimum degree, so the average degree should be high then we can just (( )), what what would be the minor the clique minor that we will get from this thing. So that is the next statement. So, this statement is Thomassen statements can be obtained from this thing. So let us look at this thing.

(Refer Slide Tim: 31:31)




Let we will take this f(r), Thomassen statement says that if the girth is greater than f(r) we can we can find a k r minor that (( )) even with minimum degree requirement 3. Initially let us say the minimum degree is just 3, d equal to 3, Diestel statement we can take d equal to 3.

(Refer Slide Time: 32:00)

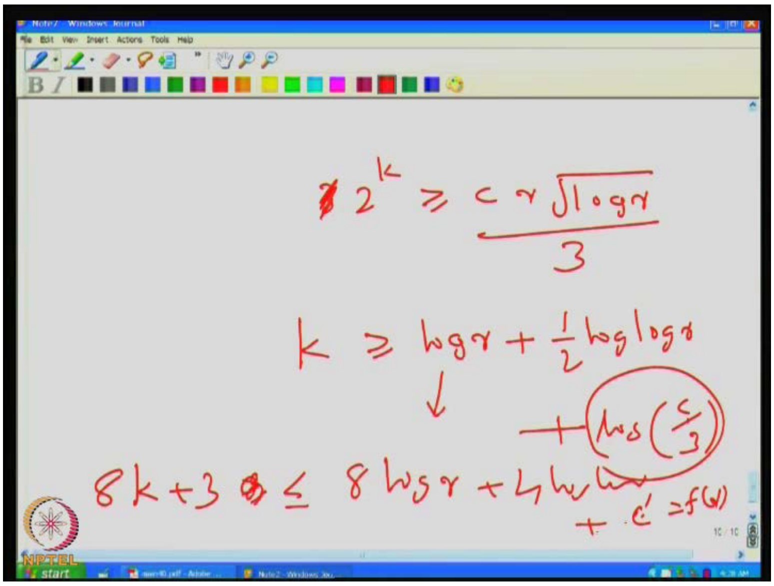



So, then we just define a k such that we will define k equal to k of r such that 2 into sorry 3 into 2 raise to 2 k is greater than or equal to 2 into… See this is that constant from Kostochka’s theorem and then r times root log r. So, this is coming from this statement, this statement this c r root log r. So, the Kostochka wanted Kostochka wanted the average degree greater than equal to c times r times log root log r right. The same function we take. And we find out that the we find out the the k smallest possible value of k such that 3 times 2 raise to k is greater than equal to c times r root log r, because why because this tells in the in this previous lemma in the Diestel lemma, our d will be put 3 now, because we are thinking about graphs with minimum degree at least 3 and d minus 1 will become 2. So, it is 3 into 2 raise to k. So, we want to find out what is that k, because and then we want to verify whether the girth requirement will be ok right.

So therefore, we will find out this thing. If this is the case we know that we do have a k r minor here right, so in H because we know by Diestel’s proof, we do have an edge with minimum degree greater than equal to 3 times 2 raise to k 2 raise to k. And then therefore we do have a because this this much this amount is greater than equal to this, this value is greater than equal to this by Kostochka’s statement, we have a k r minor here. The only question is, whether to apply Diestel’s result, we need one more thing. The girth has to be greater than equal to 8k plus 3, so here this is in Diestel’s theorem this is d, d equal to 3 we put, this is d minus 1 and this is k right. But this k also should be connected to girth,

girth should be greater than 8k; we just verify this. So, what is k? We just take a log here, this k equal to from this thing we get k equal to… So we will go to this thing.

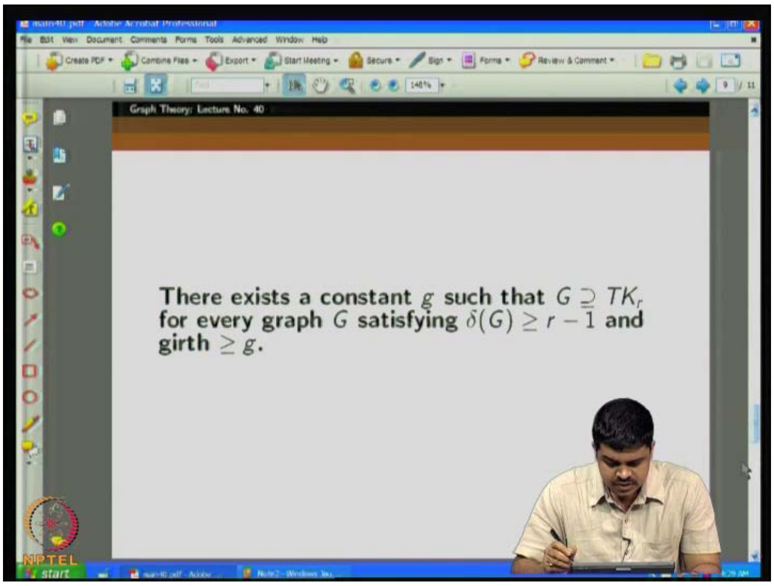
(Refer Slide Time: 34:36)



So, we have this greater than equal to c times r root log r right. So, now we can take this to this side, and then take log on both sides, we get k is the smallest value such that this is greater than log r and plus half of log log r right, but because log of root log r is half log log r minus sorry it is a plus log c by 3 right. So that is the constant her. Now, we can estimate so we can we will put some c log c value for this thing. So, approximately if I take 8k plus 3, we can easily see that this will be greater than or equal to our (( )). So, the… So, our girth this value will be like here we know 8k plus 3 is going to be at most, so we can substitute this thing 8 log r plus when we multiply this 4 log log r plus some constant right. We can fix the constant, so that it is less than this.

Now, this is indeed the f of r we have selected, so we may have to place a constant some c dash in such a way that we are above it. This function is above it. So, if you had selected the f of r, the constant here in such a way that this is bigger than 8k plus 3 as we have seen here, we can make it bigger than 8k plus 3. Send the Diestel's lemmas d can be applied and then this because our 3 into 2 raise to k is the minimum degree, then Kostochka's lemma says that we have a k r minor right that is what we get.

(Refer Slide Time: 36:56)

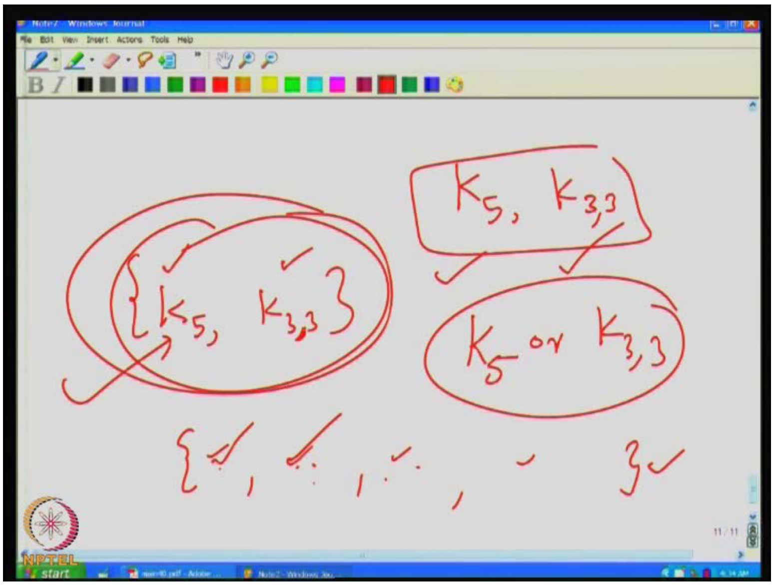



So, then so this is one example, so of showing that we have a dense minor, so here that is interesting, because the girth you just have to keep the girth high, even if you keep the minimum degree at 3 right. Then we do get large minus, we can whatever minor we want we can depending on the… So, as a function of the r k r minor we want that r we have to define the girth. So, here is another statement which as interesting.

So, if suppose if we want T k r then we know that if you want T k r here degree requirement will be, so because any each of the branch vertex has to be of degree at least r minus 1 right. So, if we just of the minimum degree requirement is same as that delta greater than equal to r minus 1. And then if the girth is a little high, we do not even have it as a function of r, we fits some constant is x is such that if the girth is greater than that constant, then we do have a T k r right. So, this is also an interesting result. Now, the so that these are some known results from the minor graph minor theory.

So, in the graph minor theory, there is an interesting results called the graph minor theorem in this, because we do not have much time now only around say 20 minutes a left around 15 minutes we will get now. So therefore, we want get into the technical details we will just quickly introduce that the concept, I thing is for instance it is much easy very easy to understand, it from the idea of this what we know about planar graphs. So, planar graphs are some kind of minor close graph, because if you if you if you take a minor of a planar graph it is again a planar graph. So that, but then we know that a planar graph can be characterized by two forbidden minors namely k5 and k3, 3.

(Refer Slide Time: 39:07)



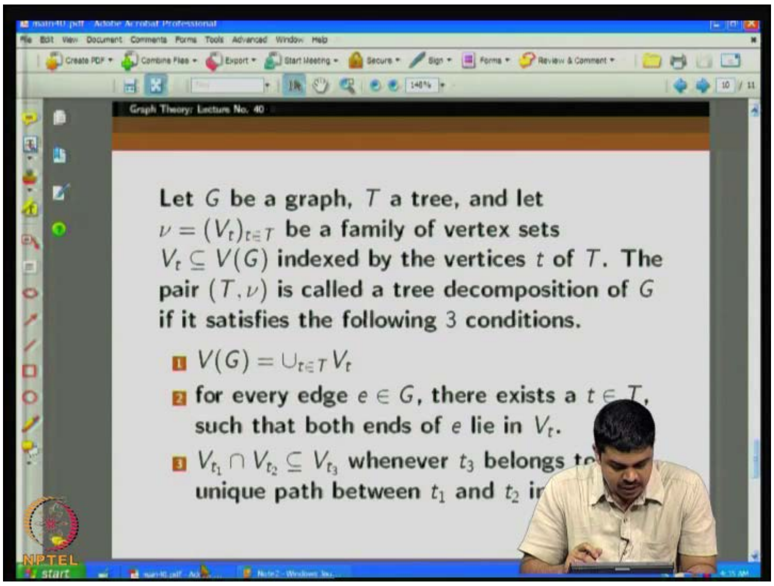

We have studied it before; that means planar graph cannot have k5, planar graph cannot have k3,3. Not only that if it is a non planar graph, they will be k5 or k3, 3 has some minor, this is what. So, the planar graph, a graph is planar graph if and only if we do not have a minor from one of this thing. So, this is listed like this k5, k3, 3. This is the set of forbidden minors for planar graphs. So, any non planar graph will get a minor either from either this or this, also from this list one minor it will have; a planar graph will not have. So, it is so happens that if if you given any minor closed class of graphs, family of graph, then we can get a list like this; not just a list when I say list we can then in list with infinite number of members in it. We can get a finite list that means a list of graphs with finite members in it such that our graph class can be characterized this way. So, the the graph class which is minor closed will be such that none of this graphs in this list finite list forbidden set of minors will be a minor of a graph in our class.

On the other hand, for any graph outside our class if our graph does not belong to this particular this particular minor close class we are talking about. Then one of this graph will be a minor of that, the same same thing like same thing like the planar graphs. So, this this is this set is called Kuratowski set, because this like imitating the Kuratowski's theorem we can say that, the that is the that is the Kuratowski set of that minors close property. So, if the property is minor closed then see none of the graphs having our minor closed property will have a minor from this list.

On the other hand, if we do not have the this minor close property for our graph, then one of this will be a minor of this. So that is a finite list of. The interesting thing is that this list is finite. In principle if you know this list, so to verify whether the graph has our property namely, this minor close property. We just have to see whether this is a minor of that, this one is a minor of that, this one is a minor of that, check for each of the members of this list whether is minor of that. If it is a minor of that and then we will if it is a minor of that we will say that yes, we do not have the property, if it is none of them is minor then we will say that we do have the property.

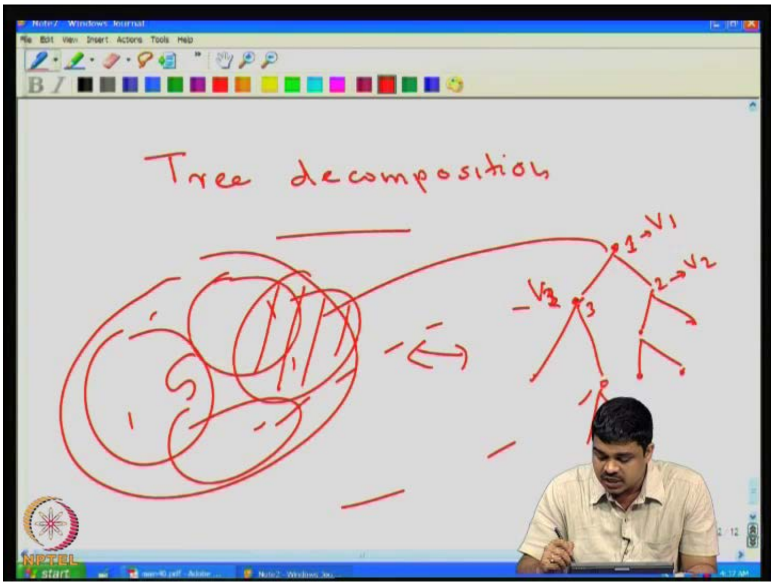
So, this theorem was proved so of case the technical statement are somewhat different that essentially we can quickly understand it in this way. So, this graph minor theorem was proved in series of papers by Roberts in (( )) and then the the several ideas were developed to prove this results. So, one of the concept which was which was developed would be of particular in that especially we are doing algorithm, computer science and so on. So that is called the tree decompositions tree decompositions.

(Refer Slide Time: 43:03)



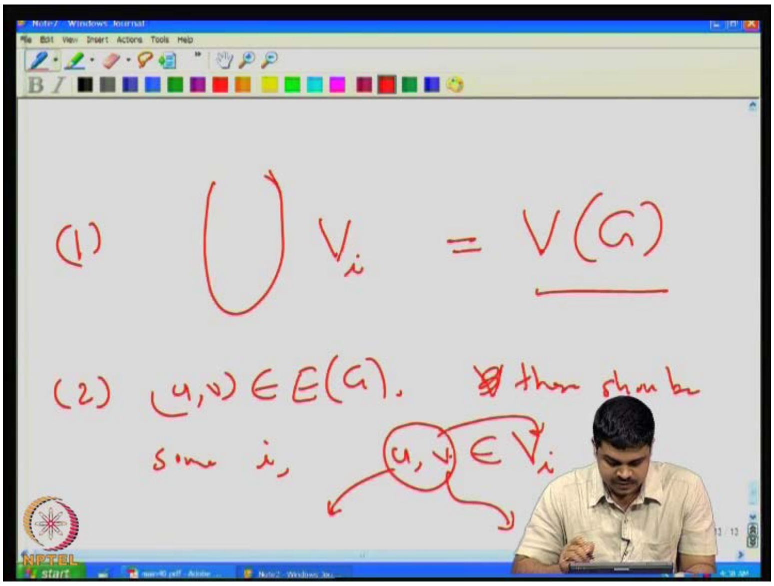

I will just give a brief introduction to this concept. So, the what is a tree decomposition? A tree decomposition is given a graph G, we are trying to visualize the graph has a tree in some sense. So, we want to collect several subsets of the vertex set of the graph. So that is what. Let v be a family of subsets of the graphs. But then the the subsets are index by the vertices of a separate tree, so I will (( )).

(Refer Slide Time: 43:41)



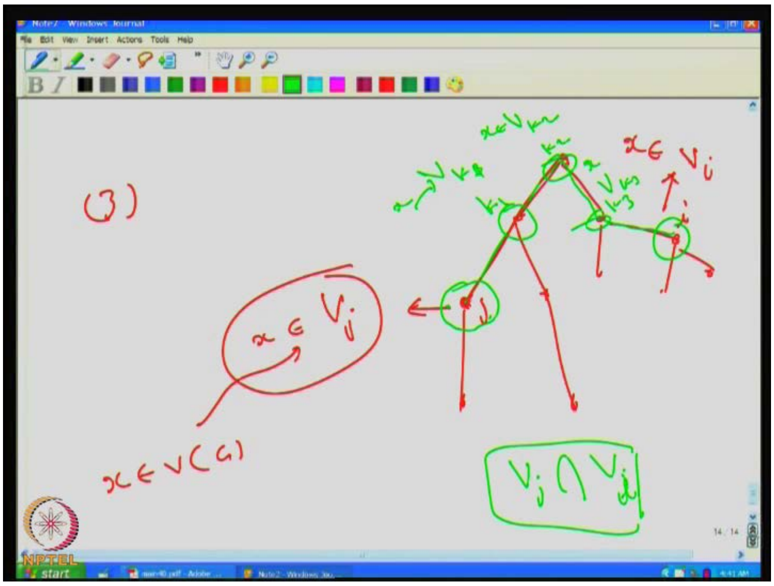

Suppose this is the graph G, now there is this vertex sets here. Now, corresponding to this, we will draw a tree some tree there exist a tree, the point is corresponding to each vertex of this tree ,we associate some subset of corresponding to this, we associate some subset of the vertices of G. So, this subsets need not to be disjoint. So, if this is 1 we may we may call v1 is associated with. This is 2 we may that v2 is associated to this. Here we may say that v3, this is 3, v3 is associated and so on; corresponding to each vertex of the tree we identify a subset so the i th vertex we say v i that is here the subset from the vertex set of G and associated with this. But we need to satisfy certain constraints. We need to make sure that this this subsets satisfy certain properties.

(Refer Slide Time: 44:42)



Like one property is if we take the union of all these subsets, we should get the total vertex set of G. It is not that there exists some vertex of G which does not belong to any of these subsets. That is not allowed to right. This is the first rule. So, each of them should go to at least one, each of them some of them can go to more than one set that is ok, but we do not allow vertex not to go any right. So that is why the union should be the total. And therefore, every edge suppose (u,v) is an edge of G, and then they should be some v, they should be then there should be some i such that both u and v together are part of V i. See there can be some V j in which u is part, but v is not. There can be some v k where v is part, but u is not, but we are only asking there should be one at least vertex set collection v i. So that both u and v take part right, we are members of. So, because we are not allowing we are not requiring that a vertex should go to exactly one vertex set right. We can go to several, so every where we need not this property, but in at least in one they should come together. That is what the second rule says.

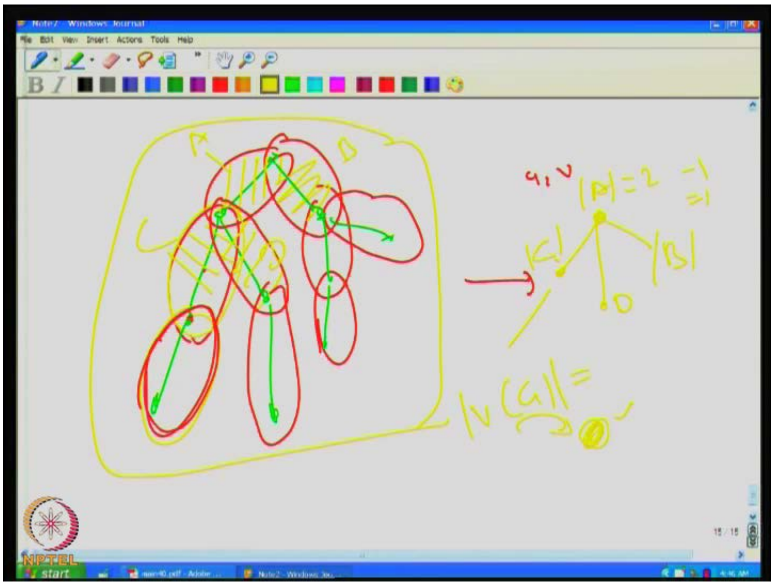
(Refer Slide Time: 46:19)




The third rule says, is more interesting rule; so it says this is our basic tree. Suppose this tree this, this is i th vertex, this is j th vertex, so here we have this certain v j corresponding to it, here we have this set v i corresponding to this. Suppose some vertex x of G belong to this v j, so the elements of v G, so is being this thing; so this is in this subset. So, suppose the same x belongs to this also when we want this x to be present then we want the to be the present in all the vertices in the… So, because we have a path here from this to this, so we want this vertex should be present, suppose this is k k 1, this is k 2, this is k 3 and so on. So, we have corresponding subsets also v k 3, so v k 2 and v k v k 1, v k 2, v k 3 and so on. So, each of them should contain x, x here, x here, so x here and so on right.

So, in other words if take the vertex set corresponding to this namely v j, and it take the vertex of corresponding to this in namely v i and v i and the take the intersection, the common vertices here and here. That intersection should be subset of the vertex of the corresponding to this, the vertex set corresponding to this, the vertex set corresponding to this and so on; all the corresponding to each vertex in this unique path that here we should have. This is this is the third property that we required. So far a graph given a graph G such a tree, and the corresponding subsets satisfying this three rules are called is together called the tree decomposition of G. So, what is this tree decomposition? This tree decomposition essentially is trying to somehow visualize the graph G in terms of the tree; that means how close is the graph to the tree, for instance if the graph itself was a tree.

(Refer Slide Time: 48:34)

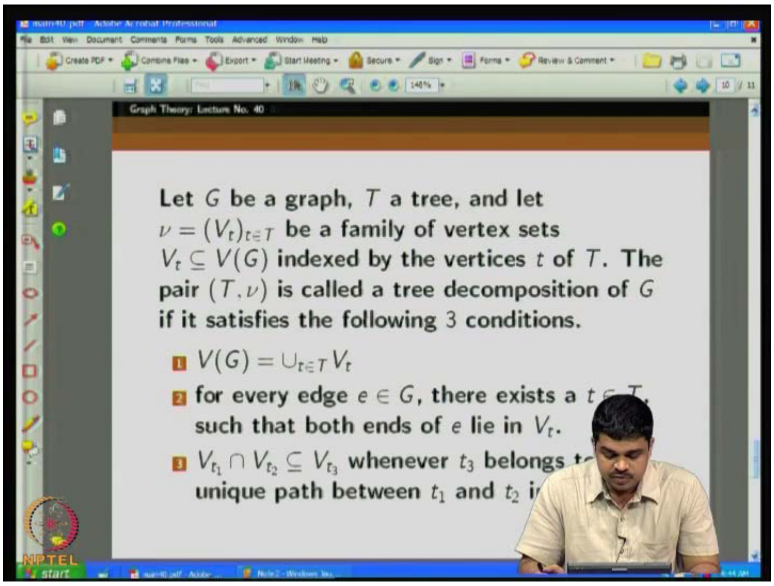


So, what we do is, so of case, how will you get a tree decomposition for this tree? So, we will define a separate tree for instance will be one vertex set, this will be another vertex set, this will be another vertex set, this will be another vertex set, this will be another vertex set, this will be another vertex set, this will be another vertex set. So when you put together one vertex corresponding to each of this sets (( )), two element sets. We can get a tree structure from that right that will be the corresponding tree.

Now, we can see that whenever… So, if all the rules are satisfied here; so every vertex is part of some subset right. And for every edge u, v so we know we have a corresponding subset here which contains both of them. Now, if you take any vertex say this vertex and this is (( )) part part of this set right; this is part of this set and this is part of this set, so of case there is they are adjacent here, so before the third property is trivially satisfied. So, this is a tree decomposition of the underlying tree.

So, the green tree is the graph in fact, so the other tree namely the tree which I drew here this tree sorry, this tree is to be constructed from this figure sorry, this tree to be constructed from this figure like this, because here corresponding to each of these these subsets I have to known, we have to put a vertex, so if this thing I put a vertex for this, so this let us say this is A, this is B, so this is C. So, I will have to put this is A, this is B, this is C, and then this is D; that means between A, A is connected to C, D, B right C, D, B. Like that we can constructed if tree from this thing. This will be the the second tree, the tree tree corresponding to the tree decomposition right. So, here each subset of the which we assign, so for instance vertex A is assign to the subset. It contains two vertices of graph and as we have verified all this sets indeed satisfies the thing.

(Refer Slide Time: 51:14)




And now it is, so why is this tree decompositions important for us? So, this is a concept developed as part of the proof of while while Robertson and Seymour were doing the proof of graph minor theorem. This concept was developed. And then it turns out to be very interesting for people algorithmic community also. It is so happens that if a graph has bounded (( )) low (( )), then we can get algorithms of very hard problems also using this tree decomposition, because it is a very good representation of the graph. Like trees are the so the Then we can define based on this concept, we can define a concept called the width of the tree. What would be the width of the tree? So, width of the tree is so among all the subsets we are associating to the vertices of the tree, we can take the biggest subsets that we have used that carnality will be width of the tree decomposition.

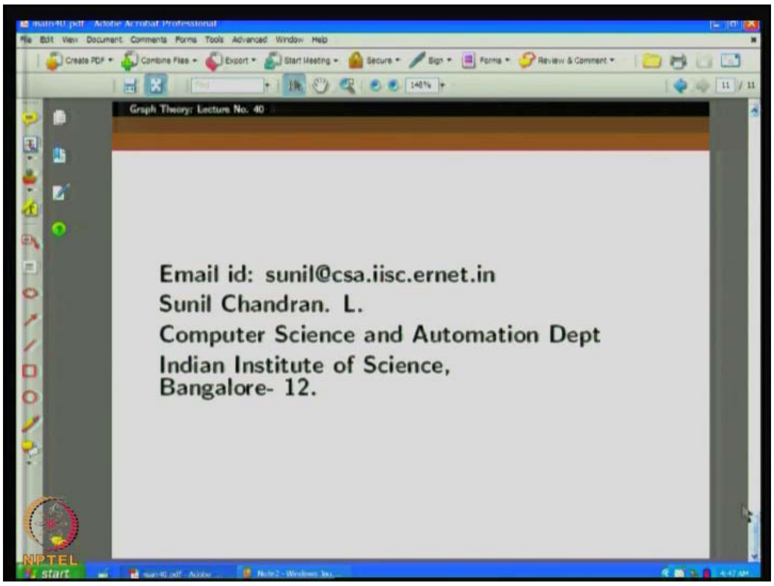
Now, of case our intention will be to minimize the width. So far a given graph we can have several possible tree decompositions. One very trivial tree decomposition is to take the entire vertex in one certain say that the my tree this and then this vertex correspond to v of G itself. This will not help us at all, because we are not another thing any structure of the graph in this thing. But here width the of the tree decomposition will be the entire vertex carnality in the entire vertex at itself.

So, but on the other hand here we can see that each of the sets are of carnality 2, the width is defined to be the maximum carnality among this thing minus 1, of case that is the technical reason, because we want the trees to get a width 1 right. So, here these are all 2, so 2 minus 1 equal to 1 will be the width of each of these decompositions. We can see that if the width is quite small that means every subsets associated to a vertex in the tree is say 2, 3 or may be a constant, it will be like the graph, you can see a conceptual tree like structure in the graph that is why we can do that. So, this structure can be made use of to design algorithms, strong the very efficient algorithms. So, this is why probably this was very interesting for the computer scientist?

So, on other than the the the possibility of designing a efficient of algorithm based on this tree decomposition. It is also useful in understanding the structure of the graph; any several structural questions also can be connected to the tree decomposition, which happens that if tree decomposition is slow, many other parameters are slow; if the tree decomposition is… So, the such connections can be in terms of tree width tree width, we can we can always associate other parameters to this tree decompositions. So, since we do not have much time, we cannot consider we cannot spend much more time on the tree decompositions and studying more.

But I would encourage the interested student to start from the text book of the Diestel and study a little bit about tree decomposition, and then there is several papers available on tree decompositions tree with which are quite useful in designing algorithms and then several in general understanding graphs. So therefore, the student can study himself. Now, what our so this I will formally declared at this course is ending here. So, this is the last class and this is the concluding lecture.

(Refer Slide Time: 55:13)




Now, if in this entire session, if you have any questions to ask, so you can contact me on this email sunil@csa.iisc.ernet.in, I am from the computer science and automation department of Indian Institute of Science, Bangalore. So, the the overall we have covered in this course, most of the important or general topics in graph theory, of case there are several more other topics, but in general like the other more popular topics like matchings, colorings, Hamiltonians cycles and minors, so not though not in detail we are touch most of them, probabilistic method, random graphs, so the so we have some other topics we having touch this probably extremely graph theory. So, we did not get much time for that and then regularity lemma such topics were not cover, but anyway so they are more advance may be.

So, we can so you can study in a later course about that or you can so Diestel's text book give some of those things. The main text book that I followed I n this course is (( )) graph theory and also Pandians moorthy graph theory book. They have a new edition that I have mention in the initial lecture. And then of case sometimes I have used (( )) graph theory text book, and also bola bash text is good for, but I have been taken much material from that for this lecture. So, for this series of lecturers, but these are all good books and then so right.

So, and about exercises, so there are several exercises for exercise problems that you can try for each of this lectures from its available in any of this text books in Diestel's books or (( )) text book or in (( )) books. So, there is no need for a for me to give as separate collection of exercises. It is better to what the student can do is after going through my lecture, the student can read the appropriate portions from the corresponding text book may be you will get a more better and more rigorous treatment in those books. So that if you read the things that will be much better, and then the exercise problems also we can solve from those books; that will completely supplement this courses. So and if you have any more questions you can send me email at this e mail address given here sunil@csa.iisc.ernet.in. Thank you.