Formal Languages and Automata Theory
Prof. Dr. K.V. Krishna
Department of Mathematics
Indian Institute of Technology, Guwahati

Module - 14
Introduction to Complexity Theory
Lecture - 05
NP-Complete Problems Part 2

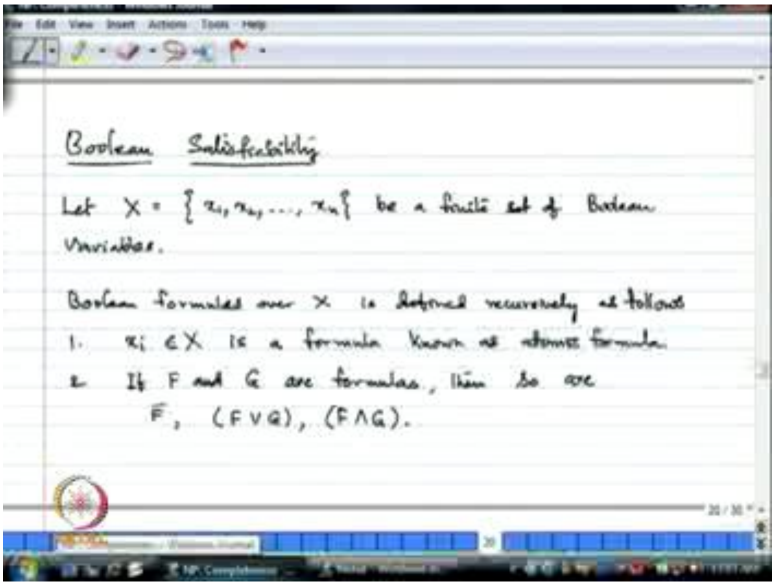
This lecture is a continuation of my previous lecture on NP completeness. So, regarding NP completeness now so far we have seen some of the problems, and we have established that they are NP complete. And here their demonstrating at a procedure to establish a problem NP complete wherever it is. And I have initiated a discussion, and introduced problem on satisfiability regarding satisfiability from Boolean formulae. So, I had mentioned that this is the first problem it was proved that it is NP complete Boolean satisfiability. The problem states that I mean general problem that given a Boolean formula whether it is satisfiable.

This is the problem, and Steven cook he has established that it is NP complete sometime around 1971, and for which he has received the prestigious Turing award. So, in this as a classroom lecture know I started giving certain reductions, and observed some of the problems are NP complete. And the first problem of course in this lectures we have observed that a quantitative you know non deterministic analog of analogous problem to halting problem, that means here what is given a non deterministic Turing machine will be given to you and a input string.

So, whether m halts on w within a given number of steps. So, though version of halting problem and comparing to this, it is a non deterministic version of course we give some time parameter also, that means within this many steps whether your non deterministic machine halts not. So, we have observed that it is NP complete, and we have observed that certain variants of these problems are also NP complete, like you know we have discussed in case of undecidability. In further, tiling problem we have observe that it was you know undecidable we have seen that, and a bounded tiling problem this is again a quantitative version so to say.

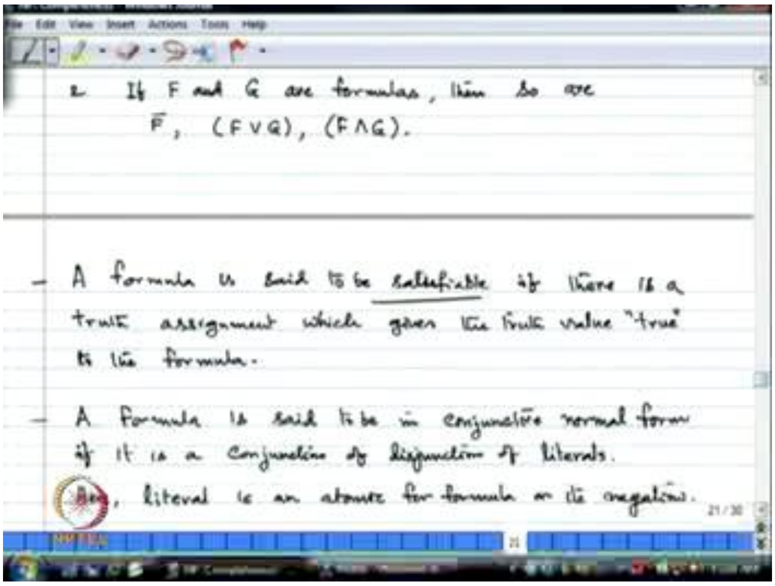
So, a tiling problem, so called bounded tiling problem that means this time you have to tie a fixed space, so a finite space. Say for example of size each side s, so s by s tiling a tiling system is a bounded tiling problem whether there is a tiling, so given a tiling system whether there is a tiling to fix to tile the region of size s, that means s by s tiling. So, we have shown that it is NP complete by reducing one of these problems, the first problem that we have shown in polynomial time to in this bounded tiling problem. So, at that point you know in that lecture I have of course introduced in a Boolean satisfiability problem also, I where I will just recap like what is satisfiability problem?

(Refer Slide Time: 03:54)



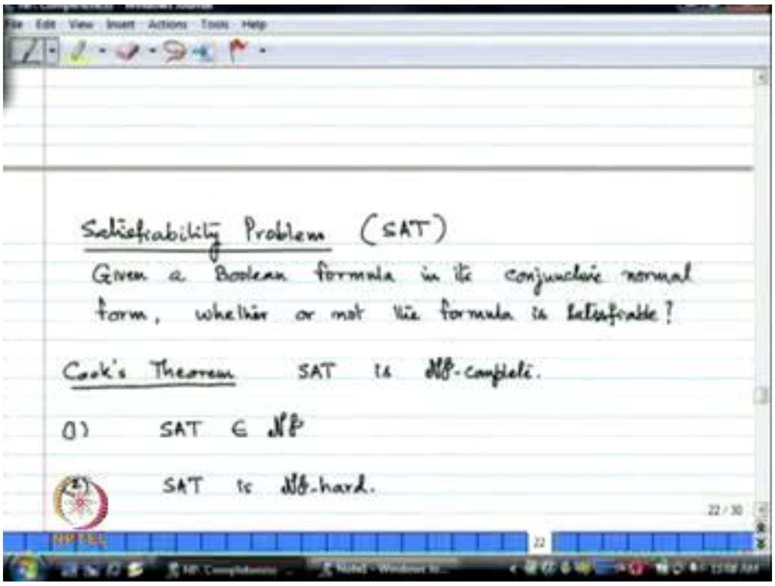
So, here the domain is a Boolean formulae, the Boolean formulae over a set of variables is essentially defined recursively using this 2 rules. So, each at a I mean each variable is a formula, so called atomic formula, and then negation of a formula, and or so called you know the disjunction of two formulas is a formula, and conjunction of 2 formulae is also a formula.

(Refer Slide Time: 04:34)



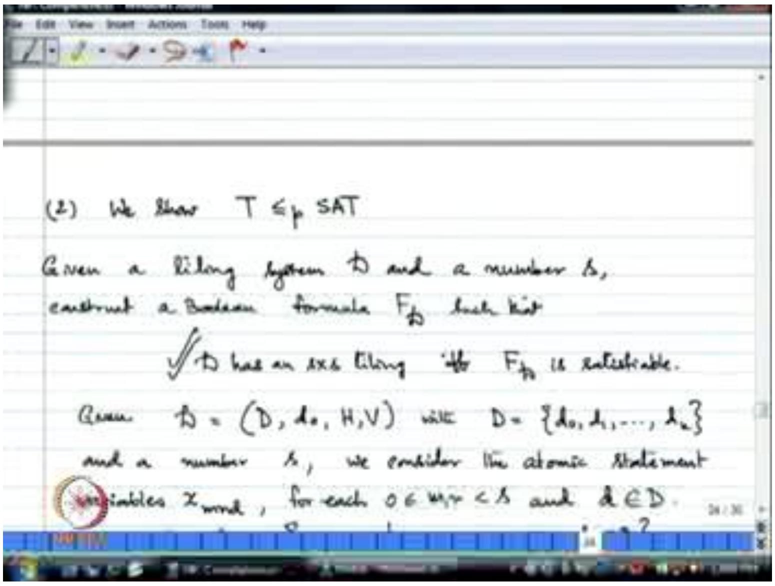
And then a formula is said to be satisfiable, if there is a truth assignment which gives truth value true to the formula, so that is the satisfiability problem. And now, the variant that we consider for the general satisfiability problem is a formula which is given in conjunctive normal form whether it is satisfiable that we consider, and the problem I am writing it a SAT, the satisfiability problem.

(Refer Slide Time: 04:56)



And the theorem Cook's theorem states that satisfiability problems SAT is NP complete. For which I ask you to observe that SAT is in NP, and in this lecture now we will observe that it is NP hard, what I am going to do here? Of course, I am in the sequence, so n naught we have shown it is a NP complete, and then I have reduce n naught to n 1 a variant of that problem. And then n 1 I have reduced to t the bounded tiling problem to show that a bounded tiling problem is NP complete. Now, what I will do? I will take bounded tiling problem the NP complete problem, and reduce the t to the satisfiability problem SAT. So, this is the change that I have considered in these lectures.

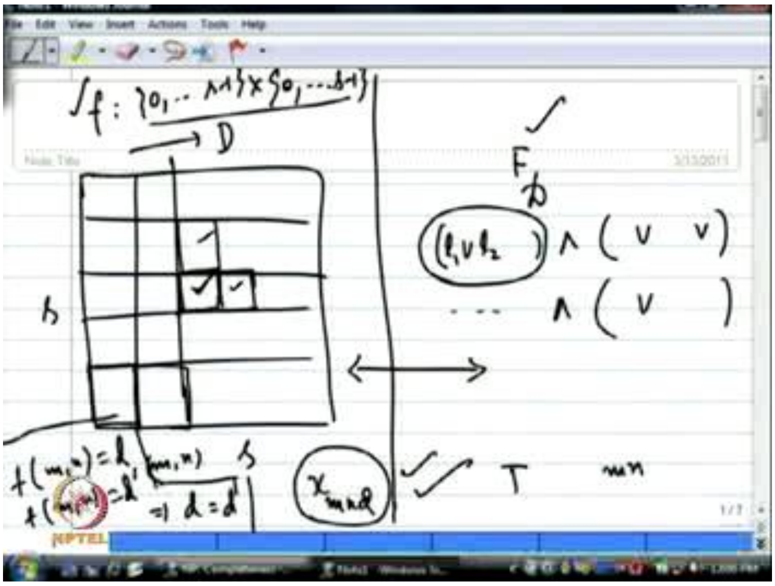
(Refer Slide Time: 05:56)



So, what, so this is what I said? The T less than equal to p SAT will show to show that this SAT is NP hard. So, what I have to do for that purpose? A given a tiling system D, and a number s, I have to construct a Boolean formula I may write F D, because this formula is based on the given system D such that D has an s by s tiling if and only if, this F D is satisfiable, this is what is essentially the condition for the reduction. So, given say D, D, d naught, H, V the tiling system with D finite set of tiles names say d naught, d 1 and so on d k, and a number s, we consider we construct a Boolean formula F D, and as the problem we have stated will be given in a conjunctive normal form.

So, I construct a formula in it is conjunctive normal form directly, so that means essentially I have to consider conjunction of disjunction of literals. So, I give you what are the disjunction of literals using the given a tiling system. And then all this put together, that means conjunction of all these disjunction of literals will be the formula F D that I am going to construct. And this F D we have to observe the corresponding to F D, I have to check this correspondence. Now, let me just give you the philosophy like of this construction, what do I do? See what is the correspondence I should have?

(Refer Slide Time: 07:31)



I should construct a formula F D. So, this is conjunction of some disjunction of some literals. So, here you will have some literals say l 1 and so on, l 2 and so on, so some disjunctions. So, this kind of formula will be constructing. And the correspondence is given tiling system, if it is having an s by s tiling, then this formula should be satisfiable and vice versa. So, that is that kind of correspondence you have to get. So, what I will do here? In this tiling system, each tile corresponded to each tile I mean fixing at a particular place, we will give a notion of you know literal here I mean variable first I will look at. Such a way that whenever I have a tiling here, the formula is satisfiable and vice versa.

So, for which what do we consider? You know fixing a tile here is the process of tiling, but the position say for example, a corner say m, n position. If I am fixing a tile, then I may you know that is the intention I will take, and consider the variables with say x, m n d you know we will consider a variable for the Boolean formulae which essentially represents in the position m n, we could fix the tile d in a tiling. So, with this intention we consider this kind of variables.

And now, if that is the intention these variable x m n d will give truth value true, I mean fixing d in m n position that means, these variable they for this atomic formula we give you the truth value true, if in the m n position we fix the tile d. If you are not fixing the tile d in m n position, then of course it should receive truth value false, so that kind of intention we consider. So, we construct a Boolean formula F D by considering the atomic statement variables x m n d, for each you know 0 less than or equal to m, n less than s, because when we are tiling the space of size s by s you know all the points m, n which we will consider less than s and d in D.

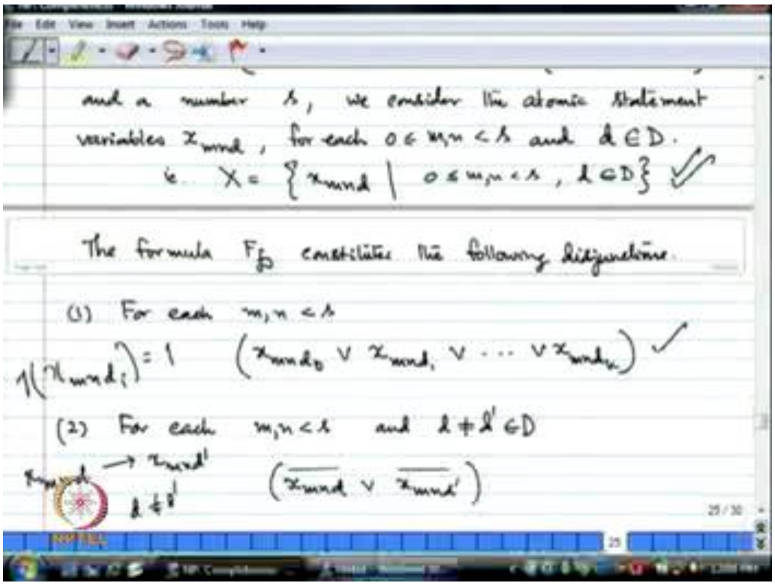
So, what is the statements variable set? This x, x m n d with this property. Now, what are all the conjunction disjunctions that I am going to consider in the for the formula F D. Now, you look at the position m, n if this has an tiling f from let me write 0 to s minus 1 cross 0 to s minus 1 to D if this is the function tiling if you have what is the meaning of that? Each position should receive a tile, each position m, n position should receive a tile, and since this is a function this should receive exactly 1 tile. And more over this tiling has to satisfy some of the I mean what are the horizontal and vertical constraints it satisfies?

If you want to fix a tile here, this and this should satisfy as vertical constraints. And similarly if you want to fix a tile horizontally side by side, then the concerning properties horizontal constraints whatever are given need to be satisfied, this is what is tiling. Now, exactly the same philosophy we translate, and prepare and construct this formula F D. So, for that purpose first you look at, what is the meaning a tile goes to this position that means among the tiles given to you d naught, d 1, d k at least 1 tile should go here, that is the first point.

And then each position should receive exactly 1 tile, because you cannot fix 2 tiles here. So, since it is a function if that has to be well defined in the element in the domain see for example, for the element m, n. If we are fixing say some d and d dash, then you know d should be equal to d dash, because at each position you can fix only 1 tile. So, this relation, because fixing at least 1 tile is important, and then to say this is a function we require this well definiteness property, because f of m, n equal to d, and f of m, n equal to d dash. Then this d should be equal to d dash, this is the thing that it will ascertain that f the assignment is a function.

And more over in the first position you require that d naught to be fixed, and the whatever the tiling that you give, what are the function f you give it has a satisfy vertical and horizontal constraints. So, all these things sort of d to be translated as Boolean formula to construct F D, and if the correspondences clear, then you can easily observe that this F D is satisfiable if and only if, you know you will have a tiling to this square of size s by s, All right. So, let us look at the disjunctions what do we give here.

(Refer Slide Time: 13:21)

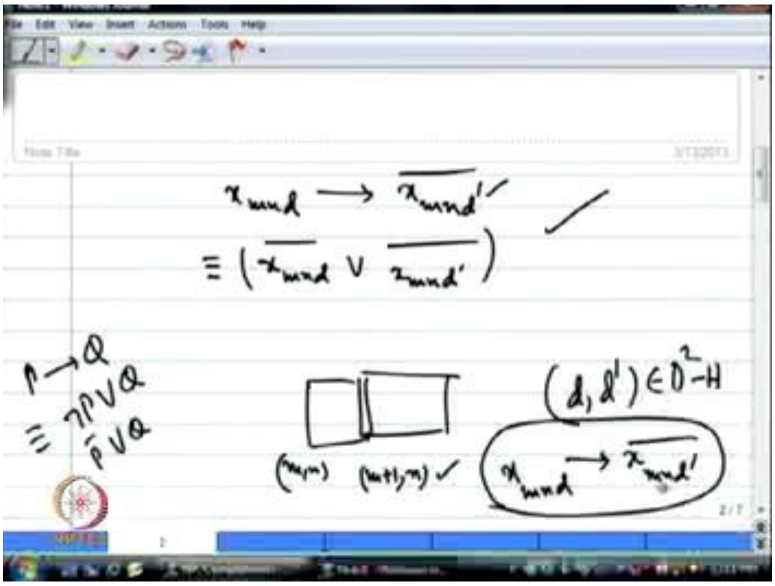


For each m, n less than s I consider this disjunction, what is the meaning of this? What is the meaning of x m n d naught that means m, n position would receive d naught or d 1 or d 2, so that is how I will read. The intension of x m n d I have already stated, the intension of the variable x m n d that we are considering in the position m, n; m, n position would receive a tile name d. So, by considering this kind of conjunction this disjunction, so this formula will be true if at least 1 of them should be true.

So, and therefore if 1 of them is given truth value true, so what our intention is that particular say for example, if this formula is true, that means x m n d i some i we have to give truth value true, otherwise you know if all of them are false this disjunction cannot be true. So, if x m n d is receiving truth value true some truth assignment T gives truth value true 1 or 0 whatever you write T. So, then what is intension in the m, n position I have received a tile d i All right. So, that is essentially captured by this formula.

And in the second place as I mentioned this has to be you know unique tile need to be received, what is the meaning of that? In m, n position if I have received a tile d then so that is represented by x m n d. Then in m, n position there should not be any other tile d dash, where d different from d dash. So, this is the condition for the well definiteness of your tiling. So, now if you look at this x m n d arrow x m n d dash for d different from d dash, All right.

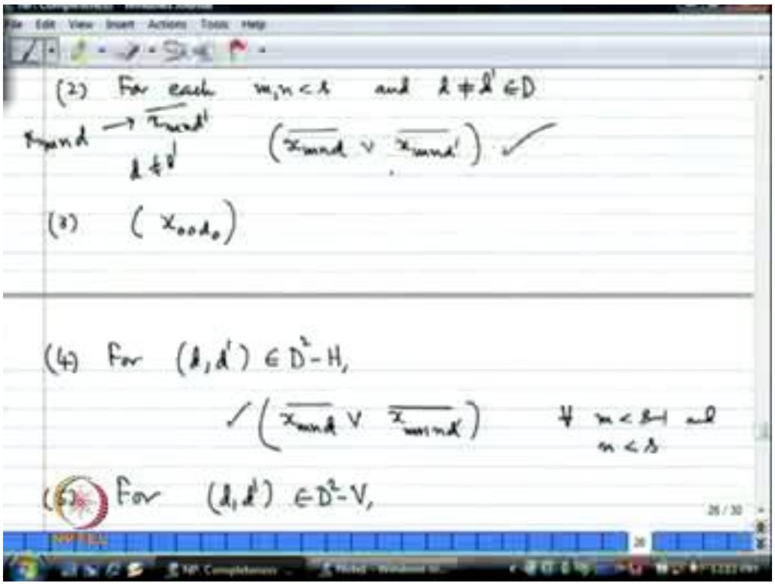
(Refer Slide Time: 15:23)



So, this can be now x m n d implies x m n d dash. So, this formula of course is equivalent to negation of I am writing with bar the negation. So, x m n d bar are, so this should not be received. So, this cannot receive x cannot receive any other tile d dash, so that means negation of this variable, so that means x m n d dash. So, if so what is the meaning of the statement m, n position if there is a tile if you are fixing d in m, n position, then m, n position cannot receive any other tile d dash.

So, cannot receive means negation of this variable. So, you know this arrow this is equivalent to this, now clearly this is in a this is a disjunction, disjunction of literals. So, this formula we consider for each m and n less than s and d different from d dash, this is a second clause second disjunction of literals that we consider.

(Refer Slide Time: 16:34)



And the third point is in 0, 0 position d naught should be there. So, I will consider this also in the formula, this is only 1 literal I have considered therefore it is a disjunction of literals, so I will consider this also. And then you look at, so horizontal constraints need to be satisfied. So, horizontal constraints are in H, so that means if you take a pair d and d dash in H, then only you know you can fix the tiles side by side horizontal constraints. So, this is the position m, n, and this is the position m plus 1, n.

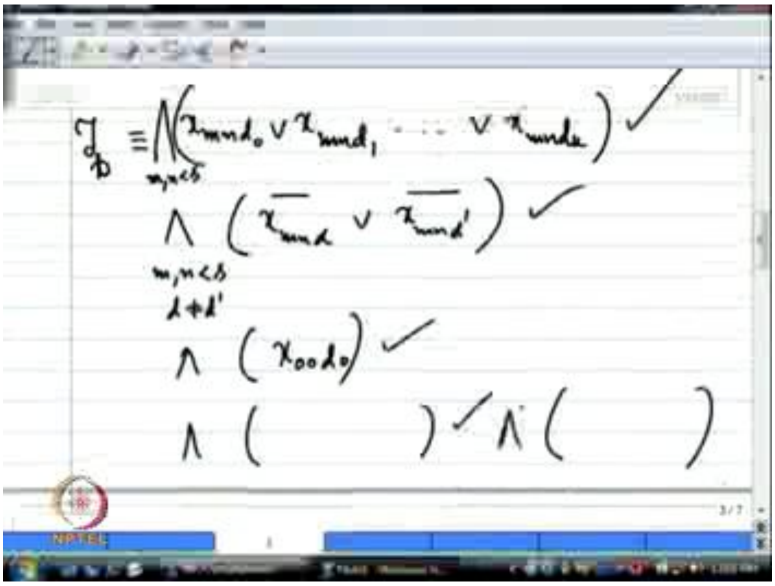
And now, if you take 2 different tiles which are not satisfying the horizontal constraints, if you take 2 tiles d and d dash which is not satisfying horizontal constraints that means, this is in D square minus H, then they cannot be fix as side by side, what is the meaning of that? If d is fixed at the position m, n that means x m n d, then you cannot fix d dash in this position m plus 1, n Position. So that means you cannot fix, that means d dash cannot be fixed in m, n, that means negation of this.

So, this again using this equivalence P arrow Q, the formula is equivalent to negation P or Q of course, I am writing here P bar or Q All right. So, I will consider this clause, I mean this disjunction of literals. Now, you look at you take any pair of tiles d and d dash which are not satisfying horizontal constraint, they cannot be put horizontally side by side, this formula can be written as disjunction of these literals. And now, since m is I am taking this way m, m plus 1 horizontal constraint I am looking for, so m is less than s minus 1, and n can be anything less than s.

Similarly, we can talk about a pair satisfying you know vertical constraints can only be fixed there, so that means, you take any 2 tiles which are not satisfying vertical constraints I will consider concerning disjunction of literals this way, this also I will take part of the formula. And now, you look at how many clauses I have considered, I mean what I mean a clause it is a disjunction of literals, how many clauses that we are considering here, you look at.

So, for each position m, n I am considering this disjunction of literals, this clause I am considering. So, how many such for each position 1 we are considering therefore, there are you know s minus s by s tiling, for s by s tiling each position need to be fixed, so the finite number of these clauses that disjunction of literals we are considering. So, for each position, so let me write this formula F D. So, for F D what we are going to consider?

(Refer Slide Time: 19:52)



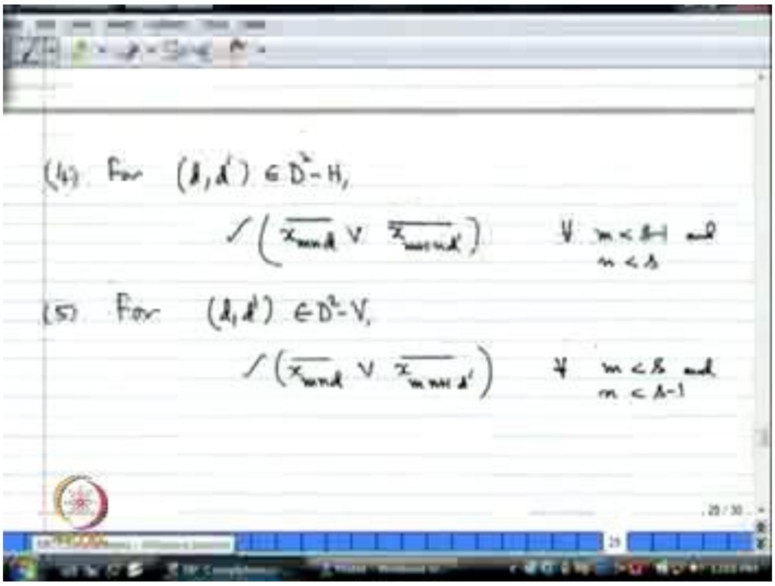
For F D is now the clauses for each position m, n. So, d naught or x m n d 1 and so on, x m n d k, so this kind of disjunctions I consider for each position that means, I may put to if you want to take m, n to be 0, 0 here, then 0 1, 0 2 and so on 0 s minus 1 likewise I have to consider, and similarly all the entire square. So, this I may simply write it as m comma n less than s. So, this is the formula that we have considered, which says that it deceives at least 1 other tile, each position deceives at least 1 other tile.

And now you look at, this receives exactly 1 tile. So, how many such for every pair d different from d dash that means, you see here how many you have k tiles you have, so that means total k square tiles. And in this k square k of them are equal, pairs d and d will come also in this. And therefore, so k square minus k for this many pairs in this, and for each position m, n I will consider this clause disjunction of literals, All right. So, again there are finitely many, so all these things need to be written down here, All right.

So, if we write if I write like this, so x m n d or x m n d dash if this for which m, n less than s, and d different from d dash. So, essentially here finitely many clauses I am writing, but I am writing into the single this thing. And next case this clause also we wanted to consider, because in 0, 0 position we have to put only d naught, so that is this guy. And the clauses concerning horizontal constraints and the clauses concerning vertical constraints, so this is what is F D? I want to consider. Now, you look at whether this is clear or not.

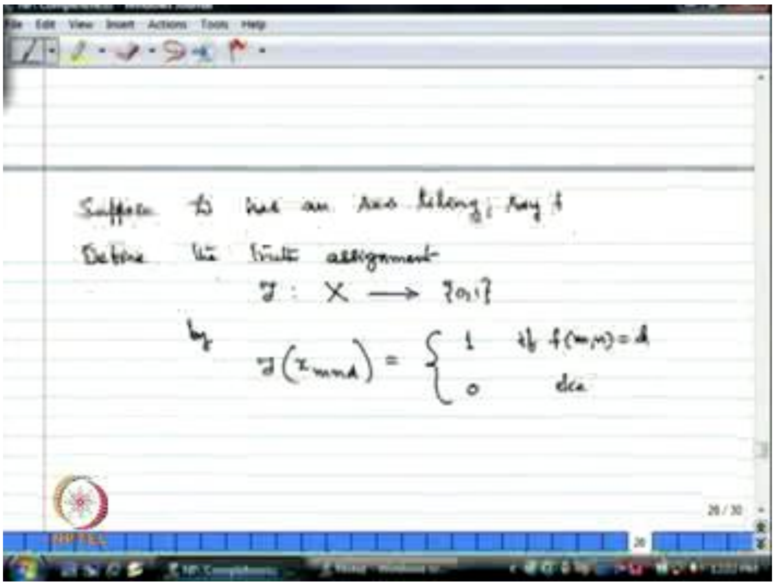
The given a tiling system D, now we have considered a set of variables with the names x m n d with the intention that, the tile d will go to the position m, n. And to have a tiling, we have to have an assignment of at least 1 tile in each position, and that is 1 which is represented by this clauses is a set of finite number of clauses here. And then you exactly receive only 1 tile to make this is a function, and then 0, 0 should receive the tile d naught with this intention we have introduced this clause, and corresponding to horizontal constraints, so we have the clauses here. So, so many I mean finitely many clauses they are to be put here, and similarly concerning this vertical constraints we are considering this.

(Refer Slide Time: 23:17)



Now, with this intention once I have constructed the formula F D, you can clearly verify that whenever you have a tiling s by s tiling, the tiling system D I can always see that there is a truth assignment which satisfies the formula F D and vice versa. I will give you here how this can be shown?

(Refer Slide Time: 23:33)

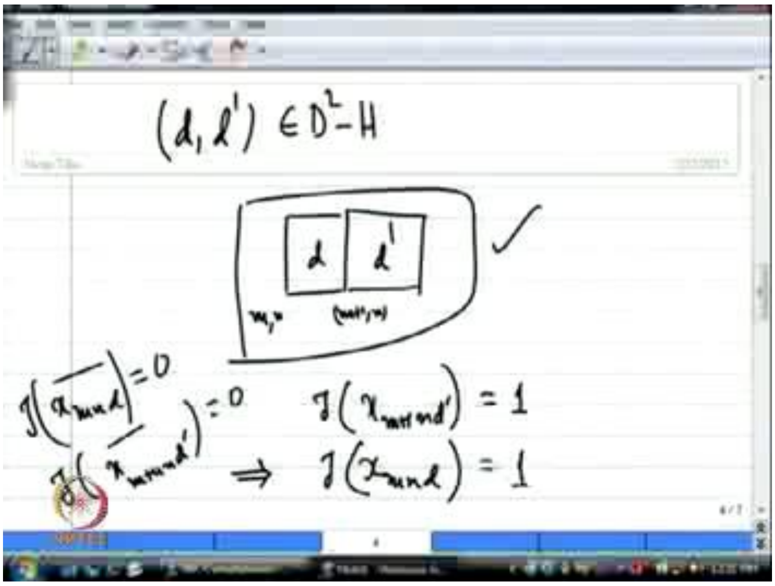


Suppose D has an s by s tiling say f. Now, define the truth assignment T from X to 0, 1, because the truth assignment you have to give from the set of variables to 0, 1, and you have to observe that this truth assignment you know. So, how do when you have an s by s tiling f, I have to say that this formula is satisfable. So, I have to give a truth assignment which satisfies the given formula, how do I define that? The intention is very clear to you. So, from that the truth assignment should be defined this way. So, the variable x m n d will be given truth value 1, whenever you fix the tile d in the m, n position.

So, that is if f (m, n) equal to d, if you are not fixing the tile d in m, n position then you simply put 0 for this, this is the truth assignment you take. Now, you look at to say F D is satisfiable, what I have to observe? Each of these clauses written down here are satisfiable. For example, for the first clause this disjunction of literals you see. Now, since it is tiling each position should receive at least 1 tile. And therefore, at least one of these variables should receive truth value true as per this assignment All right, as per this assignment.

And therefore, every clause which is defined in one every clause which is defined in 1 will be satisfied. And then since f is a function every position m, n will receiving exactly 1 tile, and therefore each clause which is defined in 2 will also be satisfied using this truth assignment. And since d naught will be placed in the position 0, 0, this clause will also be satisfied with the truth assignment given to you now. And since it is a tiling, but this tiling system it has to satisfy the horizontal constraint, and therefore this clause also be satisfied. Now, these things are little bit looking straight forward, because is this essentially representing this, now this clause if you look at.

(Refer Slide Time: 26:03)



Now, assume 2 tiles which are not satisfying horizontal constraint d and d dash, if they are fixed side by side what is going to happen? They are fixed side by side. So, what is the meaning of that in m, n position I have fixed this d in m plus 1, n position I have fixed d dash, but this is not in this, suppose this happens. If both of them suppose one of the clause is not satisfied, that means this is what is the situation? This should receive truth value 0; this also should receive truth value 0.

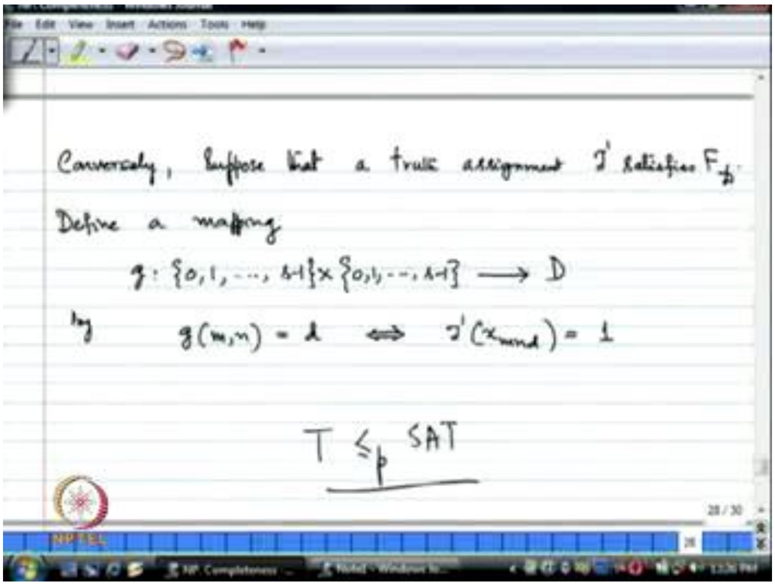
If you assume that one of this such a clauses in this list is not satisfied, that means x m d bar should receive, this should receive truth value 0. And also this should receive truth value 0, so that means x m plus 1, n d dash this also should receive, then only this is not satisfied at least 1 of the clauses in this list if it is not satisfied this is the meaning we have that. So, but this implies the truth assignment further literal x m n d will be 1, and also truth assignment for the literal x m plus 1 n d dash is also 1, that means what has happen here?

So, this are fixed side by side these are fixed side by side, but once you fix them side by side, you require this horizontal constraint need to be satisfied. But, you see that you have considered d, d dash which is in D square minus H which is not satisfying horizontal constraints, but if both of them are receiving truth value 0, then this situation is they will be fixed side by side which is not possible. And hence every clause listed in this section 4 I mean I call this section, because there are so many clauses here. So, whatever is represented in this has to be satisfied.

Similarly, every clause which is listed in this list 5 need to be satisfied, and thus you see all the clauses are satisfied. So, this clause here every clause in this list is satisfied, this is satisfied, and this is also satisfied, and all these are satisfied, and hence F D satisfied by the truth assignment T is that fine. So, what has happened now? If you assume there is a tiling for the tiling system D, we could give a truth assignment T which satisfies each and every clause or each and every that disjunction of literals which is present in the formula F D is satisfied, and hence F D is satisfied.

Therefore, we have got one direction, and in the converse also we have to observe that means, whenever this formula F D is satisfied I mean if it is satisfiable, that means if you have a truth assignment which satisfies this formula F D, we have to observe that the given system has an s by s tiling.

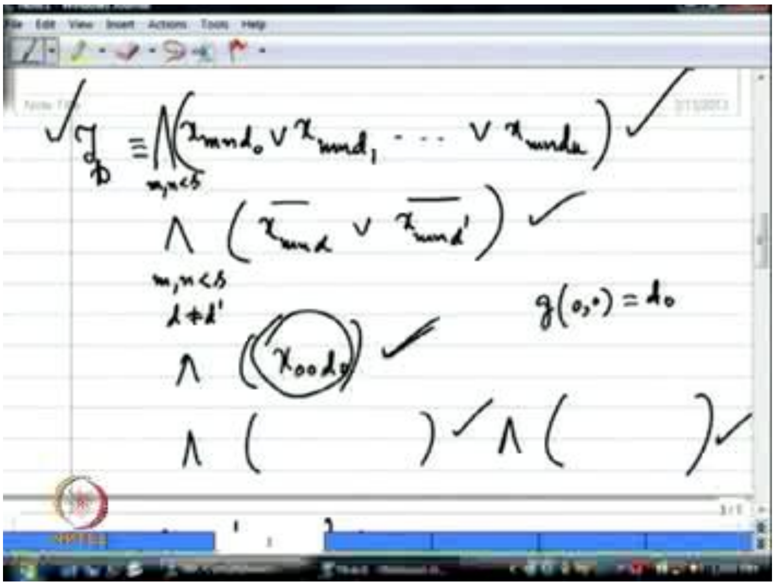
(Refer Slide Time: 29:23)



So, further purpose now conversely suppose a truth assignment T dash satisfies F D. Now, I will define a mapping g from this to this, because the tiling what I have to give that is from this set, that means for each position I have to assign tile D, how do I define that? The with this simple condition, this is what is our exactly our intension. So, whenever a variable is receiving truth value 1, then you fix the corresponding tile d in the position m, n, and whenever this happens you fix it, and you will fix only when you know that the variable receive truth value 1.

Now, we have to observe that this is a tiling; g is a tiling that means what I have to look at? This g is a function number 1, and number 2 the g satisfies horizontal and vertical constraints, and the what is called the tile d naught should go to the position 0, 0 all these things need to be verified via this function g. Now let us look at, if there is a truth assignment which satisfies the formula F D what is the meaning of that? All these clauses will be satisfied.

(Refer Slide Time: 30:35)



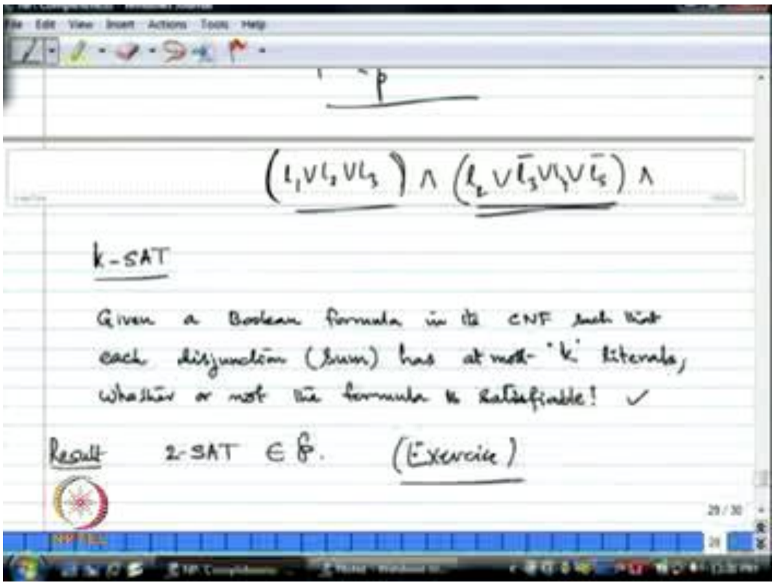
So, now this clause this disjunction of literals is satisfied, that means this particular literal which is present only one literal present in these should receive truth value true. So, if this is receiving truth value true by this construction you see 0, 0 position should receive truth value should receive the tile d naught, so that means g (0, 0) will be d naught All right. And of course, I have to observe that g is a function, since I have already explain this intention of each clause from which you can easily ascertain that this g is a function, now look at that again.

Once again I give an argument here you see every clause given in this should be satisfied, that means at least one of the variable should receive truth value true. If at least one of them is receiving truth value is true that means, the tile d i is going at least 1 tile is going to the position m, n fine. And again in the second list what we have said? If you take 2 different tiles, they cannot go in the same position at least one of them is receiving from this by this the function what we have defined here, because this is receiving truth value true each one of them is receiving truth value true means at least one of this should be true.

If at least one of them should be true that means we look at, if 1 is receiving this, then other cannot receive the tile. So, thus you can see that each of these clauses will also be since it is satisfied you can see that function g, g is a function. And more over if this clauses are satisfied, the horizontal and vertical constraints will also be satisfied. You can make this argument carefully, and observe that the function g in fact, gives you an s by s tiling.

Therefore, we have reduced in this method we have reduced that the tiling problem is less than equal to p SAT, All right. Now, I will give you some more information regarding the satisfiability problem. The satisfiability problem the general problem which we have observed that it is NP complete, now to add some more in this list of NP complete problems. There are certain variants of this satisfiability problem that we can observe that there also NP complete, you can reduce this problem satisfiability to those problems, I will just state some of them.

(Refer Slide Time: 33:21)

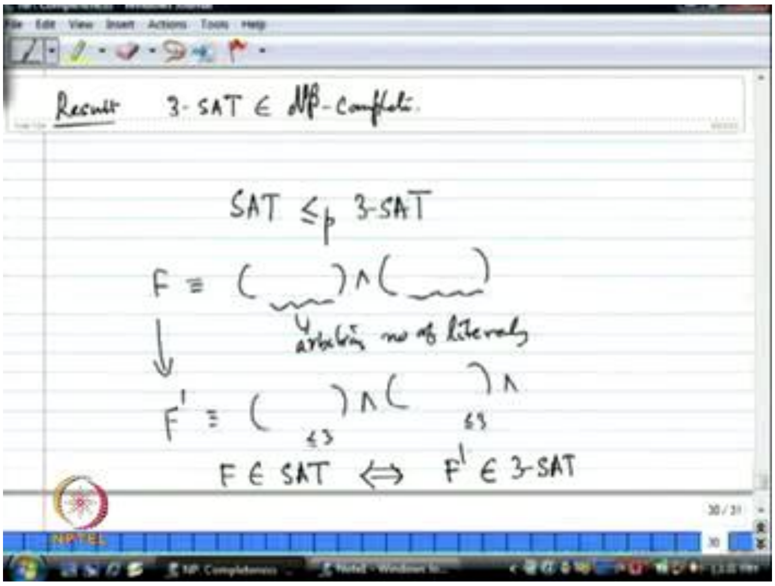


Before that first let me define the problem K-SAT, what is this? Because you are given anyway a formula in C N F conjunctive normal form. So, this time what I put a restriction is given a Boolean formula in it is C N F such that each disjunction, what are the sum that we have considered, because in C N F what do I have? This is conjunction of disjunction of literals. So, each of these disjunctions will have at most K literals. See for example, if I take K is equal to 5, this bracket within this the number of literals which appear will be at most 5. Say for example, l 1 or l 2 or l 3 say for example here l 1 say some l 2 or l 3 bar or l 4, l 5 bar is a 4 I have consider.

So, that way in a K-SAT the instance of this problem, a Boolean formula whatever that we are considering in conjunctive normal form, each of this disjunction will have at most K literals that is the instance of the problem. Now given such a formula, the problem is whether or not the given formula is satisfiable is this it is a variant of this satisfiability problem. Now, if K is equal to 2, that means each of this disjunction the clause which is present in the formula will have at most 2 literals, that means you may have 1 literal or you may have you know 2 literals.

So, that is what is essentially an instance of 2 SAT, the problem 2 SAT. We can observe that 2 SAT is in p, that means you can have a polynomial time decider to settle this problem 2 SAT. So, till that point is fine. So, you can take this is an exercise, you understand that you know the problem 2 SAT is in p, that means you have to give a polynomial time decider a procedure, but what is with K-SAT this is in p.

(Refer Slide Time: 35:38)

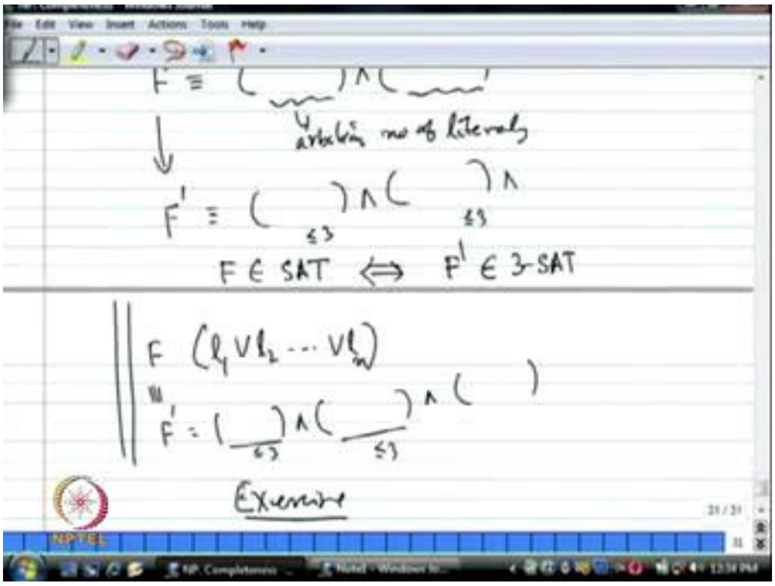


Now, if you take that K greater than or equal to 3, in fact let me just say this 3 SAT, so that means you are allow to have the literal number of literals in disjunction at most 3, that is what is essentially 3 SAT. 3 SAT we can observe that this is NP complete. So, of course in the routine manner like what we have observed for satisfaibility problem, what we could have you could have observed, that it is in NP that you have to observe. And then for NP complete, you can reduce satisfiability problem to 3 SAT. So, for this what you have to do?

So, given a disjunction of K number of I mean some arbitrary number of literals, a formula with an arbitrary number of literals over the disjunctions, what we have to reduce? You have to give an equivalent formula, by giving an equivalent formula in which at most 3 literals are present then you can reduce easily. And if you can do that with in a polynomial time procedure, then you are through, so what is that? So, you can observe that satisfiability can be reduced in polynomial time to 3 SAT. So, for that what you would you doing? You take a formula F in SAT of course, a Boolean formula, in which there is no restriction on the number of literals in the given disjunctions.

So, what are is arbitrary number of literals that you can have in this disjunctions. So, here I am not worried arbitrary number of literals, what I have to construct? I have to construct a formula say F dash given this I have to construct these. In which each disjunction will have at most 3 literals, likewise you have to construct a formula, such that what is the condition this has to be satisfied? This F is in SAT that means whenever this formula is satisfiable, then this formula should also be satisfiable, and conversely that is how you have to construct.

(Refer Slide Time: 37:59)



So, this problem can be boiled on to consider disjunction of literals some K literal, I mean let me say some n number of literals say l 1 or l 2 and so on or say some l n given a formula with this. If I can construct a formula F dash which is equivalent to this, in the sense that whenever this is satisfiable, this will be satisfiable and vice versa, and which has at most 3 literals like this. Now, you see this will fit in 3 SAT, because I am having at most 3 literals here All right, this will fit in 3 sat. And since these two are equivalent, now if you take any arbitrary formula for the arbitrary formula also just simply you will replace the appropriate you know 3 SAT instance, and then you can observe in a general case as well.

So, what is the problem now? You are given a disjunction with some n number of literal arbitrary number of literals, you have to construct an instance of 3 SAT such that the given formula F is equivalent to F dash. If you can do that, then we can see that satisfiability problem is reduce to 3 SAT, but of course this kind of thing if you can give straight forward construction which is independent of whatever that literal that you are constructing. Then you see within this you can probably have this polynomial time reduction as well. So, by this technique you know you can see that 3 satisfiability problem is NP complete.

Now, if you ask for 4 satisfiability, you see you are allowed to have at most 4 literals, now since 3 satisfiability problem is NP complete, now that problem is as hard as this problem 3 satisfiability problem, and you can observe that K-SAT is NP complete for every K greater than or equal to 3. Now, the question is how to give this? So, what we have to think that you know using this literals l 1, l 2, l n you have to give me 3 literals fitting here at most 3, which is equivalent to this. Now, you can think of this take it this as an exercise to give this reduction, which works in polynomial time, and in equivalent formula that you have to give by giving this you can observe this, All right.

So, we can discuss some more NP complete problems may be now using the satisfiability problem or 3 SAT, you just try this reducing satisfiability to 3 SAT by just giving with this kind of information whatever I had given. Now, maybe we will reduce satisfiability problem or 3 satisfiability problems too many other, optimization problems the concerning what is called decision problems, concerning decision problems have many important satisfiability problems optimization problems. And I will introduce in the next lecture, few more problems which are NP complete, and which are important in the literature.