Mathematical Logic
Prof. Arindama Singh
Department of Mathematics
Indian Institute of Technology, Madras

Lecture - 13
Adequacy of Resolution

Now, given any CNF you can just think about its clauses and then apply the single rule RPL, resolution, for propositional logic. Your aim is to show that it is unsatisfiable or not, now if it is unsatisfiable, you are confident that you will be able to derive bottom, why you are confident?

Student: Sir, this, we are removing are not p and equal to not, so those terms do not actually contribute to the satisfiability of the entire CNF's. Therefore, even neglected at the end other terms which are really not going to lead to satisfiability or rather lead to unsatisfiability, therefore, we can we say that entire (Refer time: 01:11).

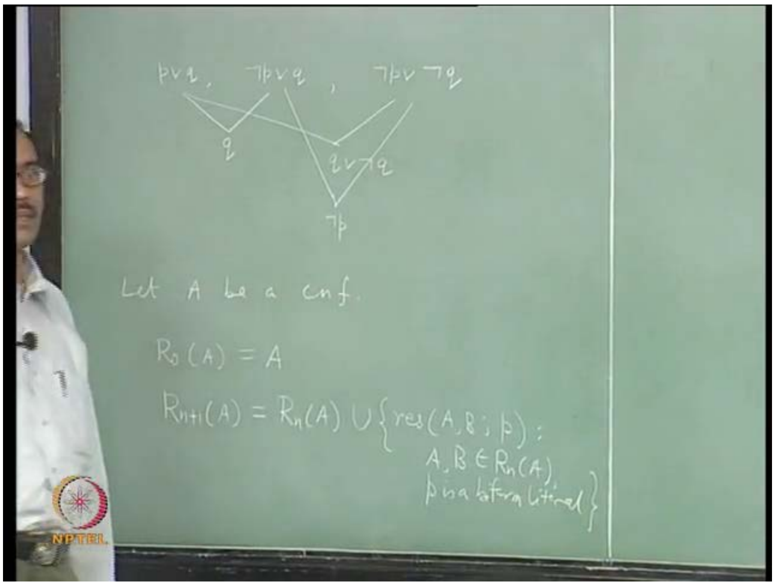
Suppose, you are not able to derive bottom can you say it is satisfiable, that is what I mean by confident, both the ways you should be able to tell it.

Student: We cannot.

Yes, they can proceed only one step, I see, I am not able to derive bottom, therefore, it is satisfiable whereas, it can be unsatisfiable, do you see the problem? It is like whether this problem is solvable or whether I can solve it. If you are not able to solve it, you say it is not solvable, that is wrong. So, what you need is a mechanical procedure whether mechanically something can be done, not only whether I can do it, but anyone. So, how to mechanize it? The resolution procedure, where we do something like, we will try to derive everything possible, very crude way.

Let us see, there should exit some mechanical procedure first, then you can modify and make it better. Suppose we start with one example, see, of these clauses p or q, say, not p or q, and say, another, not p or not q. These are the three clauses given, now you want to find out by resolution what is happening. Usually, what will you do? You proceed by taking these two, let us say, so you get q, from this the resolution taking the biform variable as p; similarly, from these you need to track what, again p.

(Refer Slide Time: 02:23)


$p \vee q$, $\neg p \vee q$, $\neg p \vee \neg q$
q
$q \vee \neg q$
$\neg p$
Let A be a cnf.
$R_0(A) = A$
$R_{n+1}(A) = R_n(A) \cup \{ res(A,B;p) :$
$A, B \in R_n(A)$
p is a biform literal $\}$

You get q or not q or, you would have taken q as a biform variable and get p or not p, that adds no information, from these two q and not q can be taken. You get not p, then, well, I am not able to derive anything, that is my position. You also say, you do not have any way to go on, but it is not possible. How to show it? Well, it is a simple example. Here, we can explore everything and say nothing is possible, but finally we will be giving an interpretation which is satisfying.

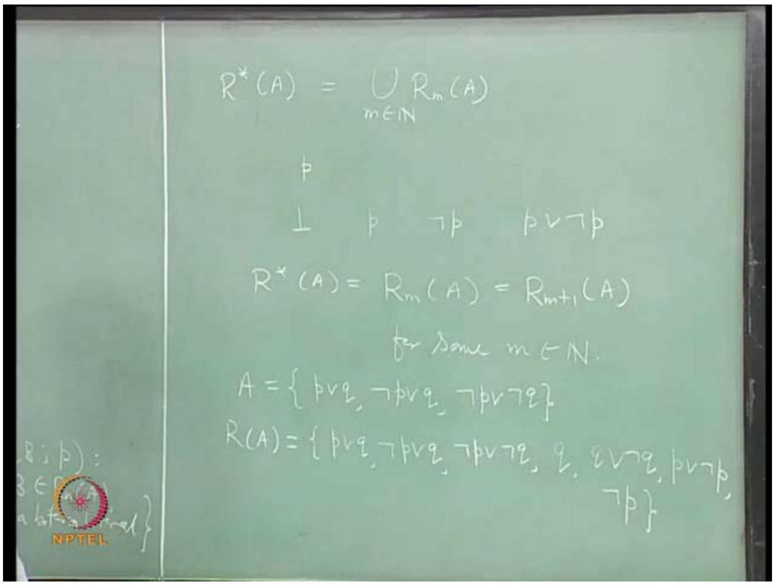
You have p and q, these two are there, so try to satisfy them, so I take q to be 1, p to be 0, these two are satisfied; and now try verifying everything. But we have to go back to semantics, that is what, it is fine. Here, we have the confidence if I am not able to go for the bottom, then I can show that it is satisfiable, somewhere, but by using semantics again. To make it mechanical, what we do is, just look at what we are doing, we have started with a set of clauses, a CNF. Let us see, then from this CNF, we go to taking resolvents, so instead of choosing some, let us take resolvents of all possible clauses, all possible pairs of clauses, then what do you do next?

Take resolvent of resolvents and continue; finally, I should say that it will terminate somewhere. Termination means it will simply repeat whatever I have already derived, it will not add anything new. I have to guarantee that it will happen, once this happens, I just check whether bottom is there or not. If bottom is there, it is unsatisfiable; if bottom is not there, the problem? It may be satisfiable. That is what I am guessing. Now it will be proved, when I give an interpretation which is a model of the CNF, may be taking into account, from all those unit clauses I have derived.

What we want really is, if I have reached that stage where I cannot add a new clause and I see that bottom is not derived, it should be satisfiable. Then, you say that resolution is a complete method. You do not need any other rules, only a single rule RPL is enough; otherwise some other rule might be required. Now, position is clear. What we are going to discuss, this is what we want to do; first formalize this procedure as resolvents, taking resolvents of resolvents and so on.

Suppose A is CNF, then what we are going to do is take resolvents of A; put them together, but then with the resolvents, you may need their, clauses of A to be resolved; not only resolvents are resolvents, but original is also fine. What we do is, will, let us say, R0 of A equal to A and then identifying Rn plus 1 of A to be Rn along with union of resolvents inside Rn of A. So, that will be resolvents of a B, with some biform variable p such that A and B are in Rn, A and p is a biform variable or biform literal. That is what we are going to do, and then finally we would like to say that I have achieved everything, no more to go.

(Refer Slide Time: 07:24)



$$R^*(A) = \bigcup_{m \in \mathbb{N}} R_m(A)$$

p

$\bot \quad p \quad \neg p \quad p \vee \neg p$

$$R^*(A) = R_m(A) = R_{m+1}(A)$$

for some $m \in \mathbb{N}$.

$A = \{ p \vee q, \neg p \vee q, \neg p \vee \neg q \}$

$R(A) = \{ p \vee q, \neg p \vee q, \neg p \vee \neg q, q, q \vee \neg q, p \vee \neg p, \neg p \}$

$B ; p)$:

$B \in$

a biform literal $\}$

So I say, it is the resolvent closure, which is simply union of all these, so theoretically it looks like this; I take all possible resolvents up to any stage what so ever. If you take R0, you have not used resolution at all, it is your original CNF. Once you take, that means you have taken resolvents, some R star you take, so that means R2, A. There you have done resolvents of resolvents, not only that, including resolvents possible along with the original CNF also. Why do you terminate? You just define, it exists, what you mean is, whether R star will terminate, whether this process will give you R star of A, but R star of A as we have defined, we take, it is mathematical.

We are not telling it is some Rm of A, you are telling it is union of all those things. What you want is, to show that R star A equal to Rm for some m, and that should be possible when say, Rm of A equal to Rm plus 1 of A. So, that is it. That should be the R star of A, because each Rm plus 1 contains Rm, by definition it is increasing. Once Rm equal to Rm plus 1, that should be equal to R star. And what is the guarantee that there is one m such that Rm equal to Rm plus 1?

There should be number, a number of, we are using some finiteness somewhere, so the finiteness comes from the CNF itself. It is a CNF, so it has some length, the number of propositional variables in it, occurring in it, is finite. Then, the number of clauses that can be generated from those propositional variables, this also finite. Suppose, I have p and q, so normally, clauses you can generate from it. Well, from p how many clauses you can generate? Suppose, I have p, a single variable, how many clauses you can derive from this one? There can be 4, no? One is bottom, trivially, another is p, another is not p, another is p or not p, all possible, you are taking all possible things. Suppose, there are n propositional variables, so there are two n literals; for each p, there is a not p. There are 2n literals, now this 2n literals can give rise to how many rows in the truth table? If you view that way, yes. If you want to form a clause using these 2n literals. Let us take, a clause has now 2n, where row consists, each one can be filled with something, let us make that places, first one is for p1; next is for p2, next is for p3, and n-th one is for pn, next one is for not p1, next one is for not p2, and so on. Now, if I give one there that means it is there; if I give 0, there it is not there, in the clause. So, one of these literals can be in the clause or may not be in a clause; so there are two possibilities. So, 2 to the power 2n maximum, so maximum these many clauses can be formed out of these literals, but then there will be some trivial clauses which we are not really taking.

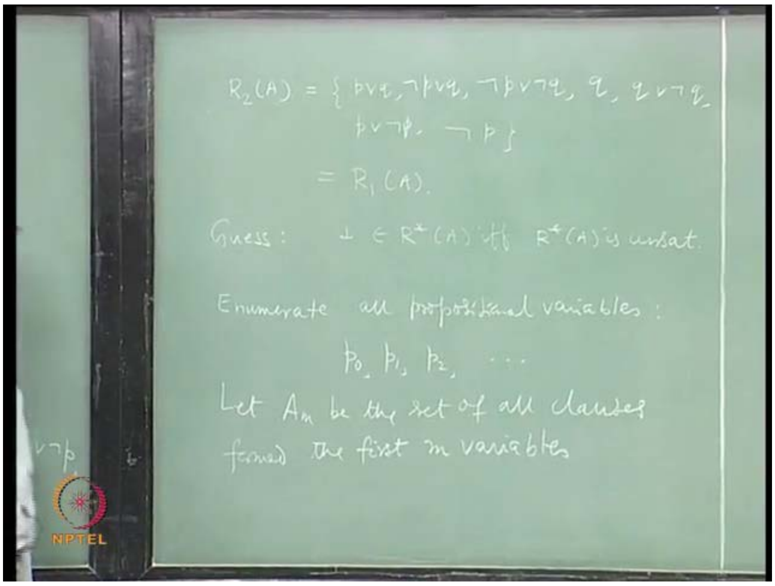
For example, this one, you may not like to have, it does not matter whether it is there or not in a CNF; so that reduces to 2 to the power 2n cases, sorry, 3 to the power n, not 2 to the power 2n, which is for 2 to the power n will have only 3 to the power n. The thing is if you take a clause, if you take a clause: either p is there or not p is there, or nobody is there. So, there are three possibilities always, so you can get only 3 to the power n, really including bottom. We are interested, so you are not deleting it, anyway whatever it is, there is only finite number of clauses possible. Now, you go for taking inductively, these sets, resolvents of resolvents, and so on, at some stage all the clauses will be over; no further, there can be no further additions can come, because that is finite. Therefore, R star A will equal to Rm of A for some m.

Student: Sir, clauses will be there.

See, truth for l, it is coming, now finally maximum of all possible, they may not be interesting to us. So, that is finite anyway. Once it is finite, you say that R star A is equal to Rm A, which is equal to Rm plus 1 of A for some m.

What happens here, we have gone up to stage 2 and stop there. So, it should terminate in R2 or R1; R1 equal to R2, it is really stage 1, it is not stage 2, it is one itself. So, R1 A should be equal to R2 A, it should be equal to R star A, that is our guess. Let us verify it; I have now A equal to p or q not p or q not p or not q. R of A will be equal to all those things and there might be more because of resolvents, so we add the resolvents. Now, starting from these we take the next one, so we get one resolvent by taking p as the biform variable, we get q, there is no other way possible. Now, one unit, there gives again with p, I get q or not q with q, I will get p, p or not p, they are not interesting, once you can delete them, but still let us go on keeping what we are getting here. These are the two possibilities, then with second and third second and first we have done; so second and third, that is, from where we are taking, let us look at these rather. Second and third, q is only biform variable, we get not, that is, all starting from A, no more is possible. Now, at this stage you can delete them, but even if you keep, it does not matter, let us go on keeping, mechanically you are doing it. Then, let us go for R2 of A, so this is A equal to R0 A, also in our notation, now it will start from R1 A, as it is. We start with p or q, not p or q, not p or not q, q, q or not q, p or not p, not p. Then start taking resolvents, all possible resolvents; first, p or q with not p or q we have taken earlier, so there is no need of two, this place, with the first clause.

(Refer Slide Time: 15:50)



$R_2(A) = \{ p \vee q, \neg p \vee q, \neg p \vee \neg q, q, q \vee \neg q, p \vee \neg p, \neg p \}$

$= R_1(A).$

Guess: $\perp \in R^*(A)$ iff $R^*(A)$ is unsat.

Enumerate all propositional variables:

$p_0, p_1, p_2, \ldots$

Let A_m be the set of all clauses formed the first m variables

So, second one not possible, third one possible with not q and q, that goes, so that gives me p or q again, fine, this q not. So, I get p or q next this one with not p, so these two go, I get p or q, next with not p, same thing happens or q is coming, q is already there. So, with the first, nothing happens, similarly with second, you have to verify it is tedious, but let us do it.

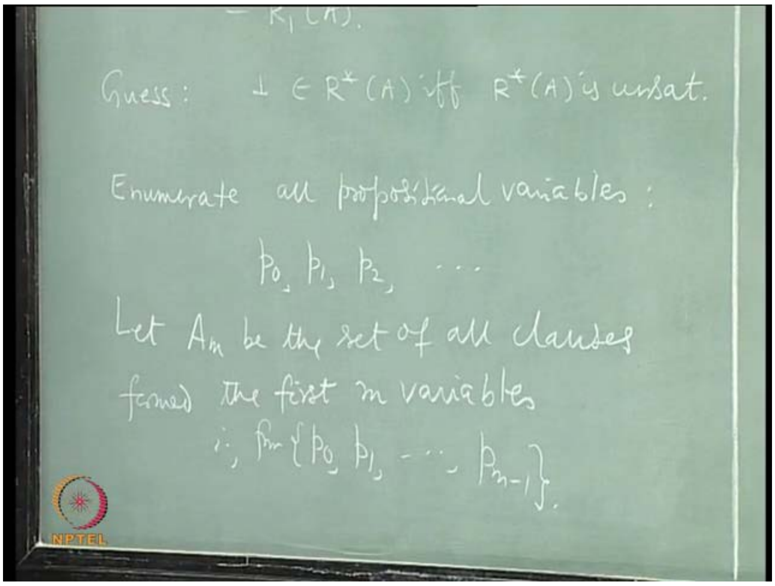
Second with third we have already done it, so second with q, not possible, second with q or not q not q. We can have not p or q not p or q, then not p or q with p or not p, not possible. It is here, already here. Next, what we do, with not p or not q, so with q that gives not p it is here; with this q and not q was, so you get not p and not q, which is that itself, and these with p gives not p or not q, that is again there, not p, not possible. Now, with q, these two will give q, these two not possible, these two not possible, then q or not q, with this, not possible, these with, not be possible. Then, these two, so not p and p goes, I get not p, that is already there. That is correct.

So, this is equal to R1 of A. That is our verification, that R star A should be equal to R1 A; it terminates there. Now, the termination is guaranteed. Our procedure says, if bottom is at all generated, it is unsatisfiable. If bottom is not generated, R star A, it is not unsatisfiable, it has to be satisfiable, so that is our guess. The guess, is bottom belongs to R star A if and only if R star A is unsatisfiable, but this is only a guess till now. One side if you can finish quickly, if bottom belongs to R star of A, then R star A is unsatisfiable; that also gives you: A is unsatisfiable.

It gives satisfiability, but because it is resolvent closure, R star of A, you have to do something more there. But that can be done because it is derived from it, because of the resolution principle any clause there is, a consequence of that, is the reason. Bottom is also a consequence of that, that is how A can be unsatisfiable. But what about the converse? If bottom is not at all generated, you must be able to show that R star A is satisfiable or A is satisfiable rather, we are interested in A, what we want.

For that, we have to do some more work. It is not easy. We will be doing something, starting from somewhere, you will not see where the connection is, but slowly we realize as we go along that there is a connection. Let us try it.

(Refer Slide Time: 22:58)



$= R_1(A).$

Guess: $\perp \in R^*(A)$ iff $R^*(A)$ is unsat.

Enumerate all propositional variables:

$p_0, p_1, p_2, \ldots$

Let A_m be the set of all clauses formed the first m variables

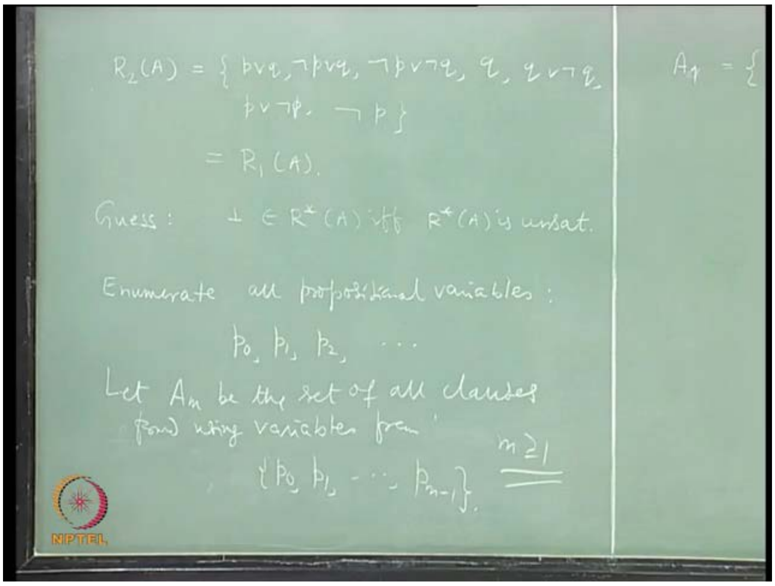
i.e., from $\{p_0, p_1, \ldots, p_{m-1}\}$.

What the starting point is, coming from somewhere we do not know, so do not be getting afraid. Now, the thing, A is a CNF. Since it is a CNF, there are only a finite number of propositional variables occurring in it; now we are going very systematically. Then we start from p0, go along, find how many are there, up to what stage the variables can be there. This is the starting point. Suppose we enumerate all the propositional variables, they are already enumerated if you look at our syntax, they are already enumerated. We have started with p0, p1 and p2 and so on. Let us write that, enumerate all propositional variables, and let us say p0, p1, p2 and so on. Let Am be the set of all clauses, so remember clauses means disjunctive clauses here, we are not specifically writing it in this context, clauses means disjunctive clauses, set of all clauses formed or using the first m variables.

The first m variables means 0 to m minus 1, that is from it A uses, from all these p0, p1 up to pm minus 1. Suppose you call it Am, all possible clauses you derive, or you are able to form not derive, using these variables p0 to pm minus 1. It is not necessary that all the variables have to be used, only a few of them I can use, that is also a clause allowed, so the other variables are absent there, that will be its interpretation.

In that case, let us see what is A0 here. For example, A0, what it can be? All clauses you can form using variable p0, you may not use at all. So, bottom, it is empty, then you can have p0, you may have not p0, you may have both, similarly next one p1, so you can have p0, you can have p1, first occurs bottom, then you can have not p0, you can have not p1, you can have p0 or not p0, you can have p0 or p1, not p0 or p1 and so on, all those you can have.

(Refer Slide Time: 26:38)



$R_2(A) = \{ p \vee q, \neg p \vee q, \neg p \vee \neg q, q, q \vee \neg q, p \vee \neg p, \neg p \}$

$= R_1(A).$

Guess: $\bot \in R^*(A)$ iff $R^*(A)$ is unsat.

Enumerate all propositional variables:

$p_0, p_1, p_2, \ldots$

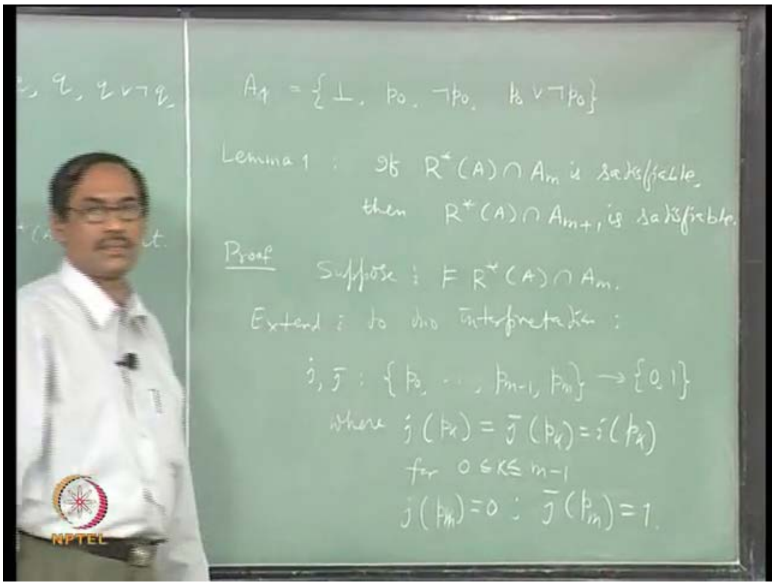
Let A_m be the set of all clauses formed using variables from $\{p_0, p_1, \ldots, p_{m-1}\}$. $\underline{\underline{m \geq 1}}$

$A_1 = \{$

Now, let us forget about that first, second and so on, that may be confusing. Let Am be the set of all clauses formed using this variables p0 to pm minus 1, but m cannot be 0. That is the problem; if you take m as 0, it will become minus 1, so this is empty again because 0 is there and we have to write 0.

We continue with, that does not matter, if you prefer this. Let us continue with that, A1 equal to this. Now, when you write this, it means m should be greater than or equal to 1, for us, now then A2 will have p0, p1. Now what do we do? First is, what we want to prove is, box.

(Refer Slide Time: 27:11)



$A_1 = \{\perp, p_0, \neg p_0, p_0 \vee \neg p_0\}$

Lemma 1 : If $R^*(A) \cap A_m$ is satisfiable, then $R^*(A) \cap A_{m+1}$ is satisfiable.

Proof Suppose $i \models R^*(A) \cap A_m$.

Extend i to two interpretations :

$j, \bar{j} : \{p_0, \ldots, p_{m-1}, p_m\} \to \{0, 1\}$

where $j(p_k) = \bar{j}(p_k) = i(p_k)$ for $0 \le k \le m-1$

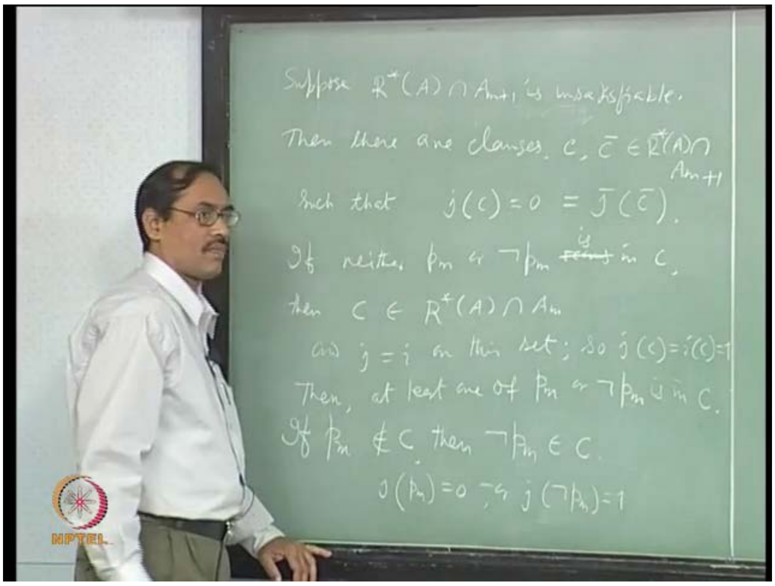
$j(p_m) = 0$; $\bar{j}(p_m) = 1$.

We want to equate both of them, so instead of going to R star A directly we will go to R star A intersection with Am, and see what happens, then finally we will generalize. That is the procedure; now you see the connection, why we were starting with this? Starting with the first m variables? Then, slowly we are increasing it so that it will be ameanable to induction. What we do is, if R star of A intersection with Am is satisfiable, then, R star of A intersection with Am plus 1 is also satisfiable, this is what we prove first. How do you proceed to show this is satisfiable?

So, there is a model of it, but this is a set of clauses, so there is a model for the set of clauses, it is a CNF of course. Now instead this set of clauses there is model, it satisfies each of the clauses simultaneously, that is what it says. Suppose i satisfies R star A intersection Am, so what is the difference between Am and R star A intersection Am plus 1? Notation, we have to be careful; now, pm, first note there or leave here in R star A intersection Am, possibly up to pm minus 1 are there, pm is the extra one. There can be some more clauses, now when you think of this interpretation i, it is not defined. Till now, for the variable pm, but when you want to make a model for R star A intersection Am plus 1, you need to define some value for this pm. So, we will extend this interpretation i to another interpretation taking care of this variable pm, now, i to, two interpretations as follows. What we do, let us call them the j and j bar, they are extensions of i.

They will be defined from the set p0 to pm, now i was defined only up to pm minus 1, now pm, we have to take care. There is a possibility of extending by assigning pm to 0 or pm to 1, which is why we are starting with two extensions, we do not know which one will work. They are same, that is what, we are writing here i of pk for 0 less than or equal to 1 less than or equal to m minus 1, and j of pk equal to 0, j bar of, sorry pm last one, p m equal to 1. These are the two extensions possible. Our contention is that, which one of this will work for R star A intersection Am plus 1, we do not know. So you proceed by proof by contradiction. We say, suppose that R star A intersection Am plus 1 is unsatisfiable.

(Refer Slide Time: 32:07)



Suppose $R^*(A) \cap A_{m+1}$ is unsatisfiable.

Then there are clauses $C, \bar{C} \in R^*(A) \cap A_{m+1}$

such that $j(C) = 0 = \bar{j}(\bar{C})$.

If neither p_m or $\neg p_m$ ~~occurs~~ is in C,

then $C \in R^*(A) \cap A_m$

and $j = i$ on this set; so $j(C) = i(C) = 1$

Then, at least one of p_m or $\neg p_m$ is in C.

If $p_m \notin C$ then $\neg p_m \in C$.

$j(p_m) = 0$, or $j(\neg p_m) = 1$

Suppose R star of A intersection Am plus 1 is unsatisfiable. If it is unsatisfiable, then neither j nor j bar is a model; it does not have a model, and we have two here j and j bar, neither of them is a model. Why neither of them is a model? It is a set of clauses. Some clause is not satisfied, so that means j does not satisfy some clause, j bar does not satisfy some clause, may not be same. Then there are clauses C bar in R star A intersection with Am plus 1, such that, such that j of C equal to 0, also j bar of C bar equal to 0.

They are not satisfied, so they are assigned to 0, which is why this is not happening, this is not satisfiable. Now, you know that j of C equal to 0, j bar of C bar equal to 0. Now, look at pm; pm may be occurring in C or may not be occurring in C, similarly, not pm as a literal may be occurring in C may not be occurring in C, now can it happen that neither pm nor not pm occurs in C?

Then, both j and j bar will be like I, because they agree with i on R star A intersection Am. So, let us write it, that one, if neither pm not nor pm occurs in C, then C belongs R star A intersection Am, and j is equal to i on this set. So, j of C should be equal to i of C equal to 1, right, contradicting j of C equal to 0. So, what is your conclusion? At least one of them occurs. Then, at least one of pm or not pm is in C, it will not belong to C, C is a set, now it is a clause. This is what I do. It means, it is not really occurrence, it is really belongs to C or not. Let us write it out, better write is in C, as it occurs. This means if not pm occurs, we may say pm also occurs, we do not mean that; we mean that either pm belongs to it or not p belongs to it, that way. If neither belongs then there is a contradiction, therefore at least one of them belongs to the set.

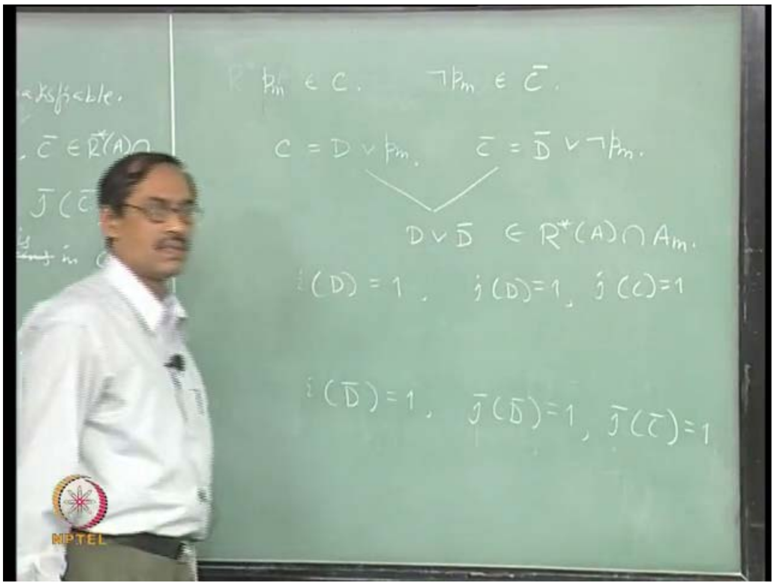
If neither belongs, then C has no occurrence of p at all; then C belongs to R star A intersection Am; on that i is a model. So, i of C equal to 1 and i and j, so j of C will be equal to 1, but j of C equal to 0. I will repeat. This variable pm is not in R star A intersection Am, it is in Am plus 1. Our notation may create problem, we have to change the notation, we change the notation pm occurs in Am or not in Am.

Once you accept C belongs to R star a intersection Am, and i is a model of R star a intersection Am, i of C equal to 1. That is your assumption; i is a model of R star A intersection Am, so i satisfies C, i of C equal to 1, but i and j agree on all the variables up to pm minus 1. So, j of C is also equal to 1, but j of C equal to 0; that is the contradiction. Therefore, at least one of pm or not pm belongs to C.

We go to the next stage. Can it happen that not pm belongs to C, that is, pm does not belong to C? Let us see. If pm does not belongs to C, then, not p belongs to C. Why? Because at least one of them occurs, at least one of them belongs to C. If pm does not belong to C, then not pm belongs to C; if not pm belongs to C, then what happens?

Let us find out j. j of pm is 0, so j of not pm is 1, it is a clause, so j of C has to be 1, whatever value of pm is, does not matter even if have D, does not matter. Now, not D; j of C has to be 1, but j of C is 0, that is the contradiction. So this cannot happen. Therefore, pm belongs to C. Similarly, what happens for not pm, so we get pm belongs to C, similarly not pm belongs to C bar.

(Refer Slide Time: 42:38)



...satisfiable.
$\bar{C} \in R^*(A) \cap$
$\bar{j}(\bar{C}$
... in C
$p_m \in C$. $\neg p_m \in \bar{C}$.
$C = D \vee p_m$. $\bar{C} = \bar{D} \vee \neg p_m$.
$D \vee \bar{D} \in R^*(A) \cap A_m$.
$i(D) = 1$. $j(D)=1$, $j(C)=1$
$i(\bar{D})=1$. $\bar{j}(\bar{D})=1$, $\bar{j}(\bar{C})=1$

.

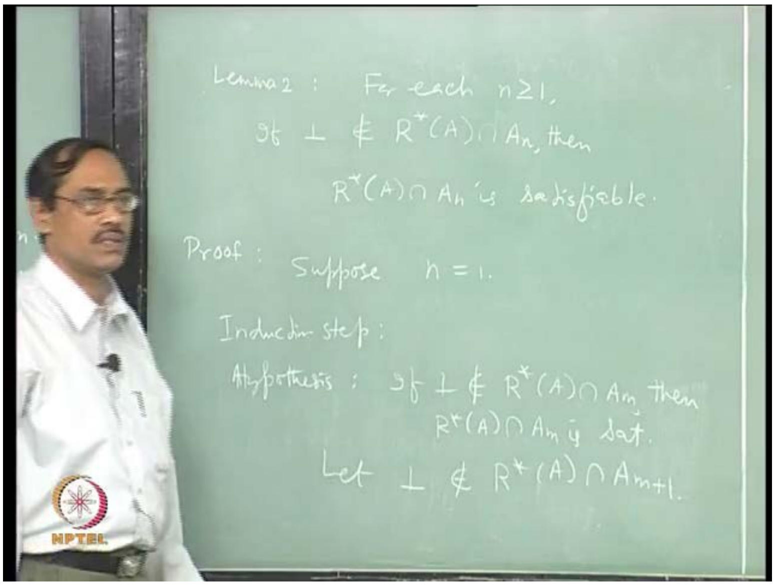
Now we have to repeat really two stages arguments. First, start with: if neither pm nor not pm is in C bar, that is a contradiction. Next, we start with: if not pm does not belong to C bar, then, j bar, you have to concern with, j bar of not pm is 0 so that of j bar of pm is not 1, therefore, j bar of C bar will be 1, but j bar of C bar is 0. That is the contradiction. That is why, not p must be in C bar. Then we can write C equal to D or pm, C bar or D bar or not pm, both of them belong to R star A intersection Am plus 1, fine. Now take that resolution, that gives D or D bar.

Now, D or D bar belongs to R star A intersection Am. Why? Because resolution completeness is there, you are taking resolution closure, R star. So, this is in R star A and this is not in Am plus 1. Well it is, but we cannot say it is in Am. Am is a subset of Am, in that sense. This belongs to R star A intersection Am, fine? Is that so? Since it belongs to, i satisfies, so i of D may be equal to 1 or i of D bar may be equal to 1, because i satisfies this, i is a model of R star A intersection Am.

This belongs to that set, so i is a model of this, therefore i of D is equal to 1 or i of D bar is equal to 1, 1 here, this has to happen. Now, if i of D equal to 1, then what happens? What about C? We cannot say j of, well, we can take one more step. If i of D equal to 1, then, j of D equal to 1, because i and j agree up to pm minus 1, so this gives j of D equal to 1. Then we have j of c equal to 1, a contradiction. But j of C is equal to 0. If i of D bar equal to 1, then we say j bar of D bar is 1, because i and j bar agree up to pm minus 1.

Then, we conclude j bar of c bar equal to 1, that is also a contradiction. So, that is all. In every case we are getting a contradiction, whatever way we are going to proceed. Why the contradiction? Because it is proved. We are not explicitly constructing which one of them will work, but say that one of them has to. It can be unsatisfiable, that is what it says.

(Refer Slide Time: 46:17)



Lemma 2 : For each $n \geq 1$,
If $\perp \notin R^*(A) \cap A_n$, then
$R^*(A) \cap A_n$ is satisfiable.

Proof : Suppose $n = 1$.

Induction step :

Hypothesis : If $\perp \notin R^*(A) \cap A_m$, then
$R^*(A) \cap A_m$ is sat.

Let $\perp \notin R^*(A) \cap A_{m+1}$.

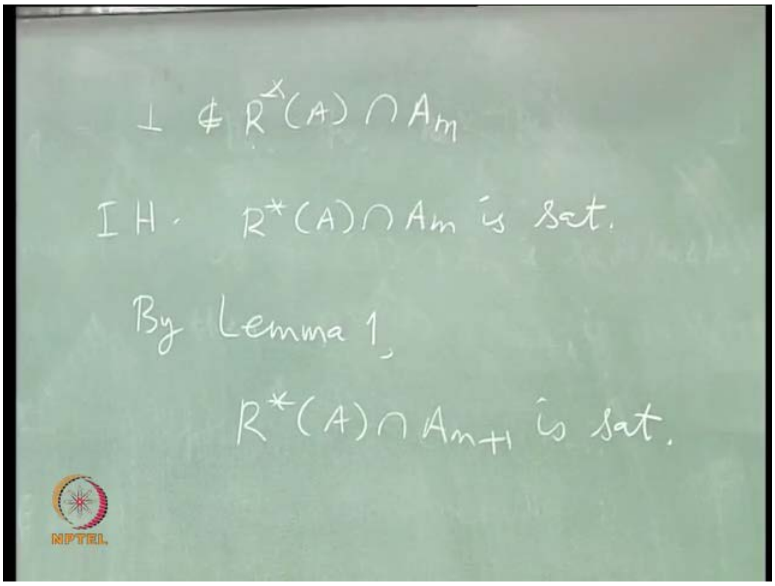
Let us take the second step. It says, though we are not proving for R star A directly, for R star A intersection An, we can do something, a similar thing. We wanted for R star A here. We show that it is for a subset, not for the whole R star A. If box is not generated up to n-th stage, then we can say that is not satisfiable, we cannot exactly, n-th stage.

Let us see how do you prove it. Suppose, but how do you prove this? It does not look straight forward; no. Let us prove it by induction on n. So, first we can take n equal to 1; in that case, let us verify if box does not belong to this, then R star A intersection A1 is satisfiable, this is what you have to verify. Then what can be R star A intersection A1?

It depends on, of course, what this A is, ha, that does not matter. Let us find A1 itself, A1 is there, but R star A intersection A1 does not contain bottom. So, R star A intersection A1 can have anything, but not this, it can contain other things and it can be a subset of this, it is intersection. It can be a subset of this, any subset of this, now whatever subset, you take it is satisfiable, the bottom does not belong to it. Whatever subset of this set you take, that is satisfiable, just check. Can p0, not p0 be there both. If both are there, then by resolution, bottom will be there; that is the only case you have to consider. But it is R star A, you are taking not only A1, you are taking R star A. Resolvents were also there. Now, if both p0 and not p0 are there, then bottom also will be there, that is not permissible. Therefore, both p0 and not p0 are not there. If not there, then you consider for all subsets possible, one is p0, another is not p0, singletons, another is singleton p0 or not p0. Then, other combinations, all of them are satisfiable, so this case is over; done for the basis step.

Induction step, so induction step our assumption. Let us write it as induction hypothesis: this says if box does not belong to, if bottom does not belong to R star A intersection Am, then R star A intersection Am is satisfiable; this is our assumption. Now, you want to prove, if box does not belong to R star A intersection Am plus 1, then R star A intersection Am plus 1 is satisfiable, fine. For m plus 1, we want to prove, so we have to start with this one.

(Refer Slide Time: 51:44)



$\perp \notin R^{*}(A) \cap A_m$

I H. $R^{*}(A) \cap A_m$ is sat.

By Lemma 1,

$R^{*}(A) \cap A_{m+1}$ is sat.

So, let this happen, that empty clause does not belong to R star a intersection Am plus 1. We want to show R star A intersection Am plus 1 is satisfiable. This is what we want to show. So how do we show that? See, this R star A intersection Am plus 1 is a superset of R star A intersection Am, it contains that, therefore if box does not belongs to this bigger set, box does, bottom does not belong to R star A intersection Am.

Now, you see that bottom does not belong to R star, sorry A intersection Am. The reason is, R star A intersection Am is a subset of R star A intersection Am plus 1. So, bottom does not belong to it. If bottom does not belong to it, by induction hypothesis, this is satisfiable, we will just finish; then by Lemma 1, R star A intersection Am plus 1 is satisfiable, over. That is what we wanted; it is a simple step. Once you finished Lemma 1, it is simple. All that we have done is, we have done it up to An, if bottom does not belong to R star A intersection An, then, this is satisfiable; this we have to generalize to R star A, directly now, not basing on this Am.