INDIAN INSTITUTE OF TECHNOLOGY GUWAHATI

NPTEL

NPTEL ONLINE CERTIFICATION COURSE
An Initiative of MHRD

VLSI Design, Verification & Test

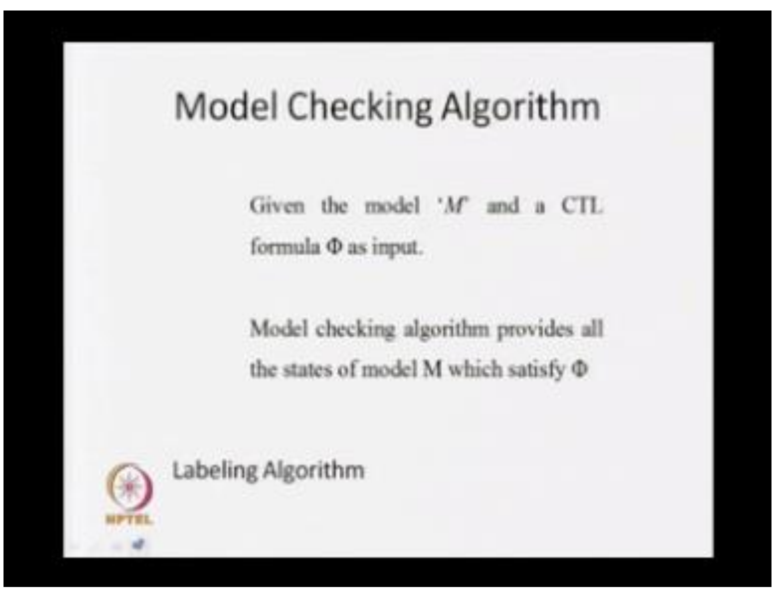
Prof. Jatindra Kr. Deka
Department of CSE
IIT Guwahati

Module V: Verification Technique

Lecture III: Model Checking Algorithms

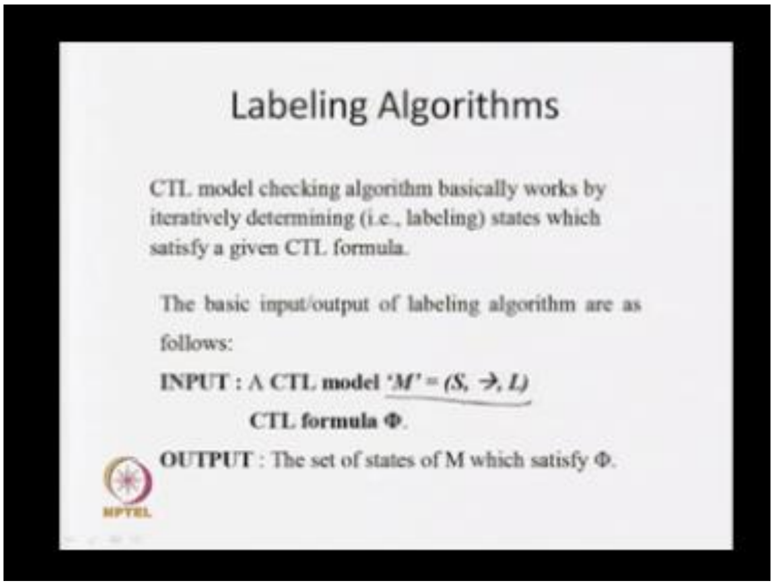
So in last class we have discussed about model checking algorithms okay, and what is this model checking algorithm basically we have seen in some sort of labeling algorithms. We are going to take the Kripke structure we will take the CTL formula and after that we will run our labeling algorithm to label each and every state with the formula which are true in this particular state.

(Refer Slide Time: 00:53)



So basically that is why we are saying that we are going to give a model M and a CTL formula for it. So model checking algorithm provides all state of model M which satisfy ϕ, so it is some sort of your labeling algorithm.

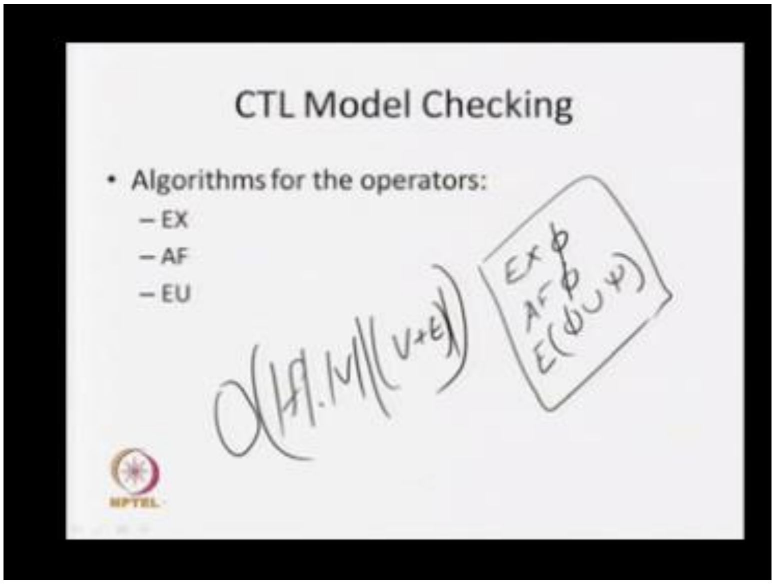
(Refer Slide Time: 01:04)



So what basically we have in this particular labeling algorithm the input to this particular algorithm is your model M which is having set of state S the transition relation and the labeling function. So already we have discussed all those issues in our kripke structure and along with that we are giving a formula ϕ, now we will check whether this formula ϕ is true and will be smaller and basically it is going to be the set of states of N which satisfies ϕ.

So it is going to return a set of state where it is true. So in last class we have seen three algorithms basically for EX, AF and EU.

(Refer Slide Time: 01:45)



EX is basically EXϕ that means there exist a path in next state ϕ hold AFϕ in all ϕ in future ϕ holds and E ϕ U ψ there exist a path so the state ϕ remains true until ψ becomes true. And we have seen that these three are the adequate set of temporal operators by which we can derive the other CTL operators, because we are having four temporal operators along with that two path quantifier all together we are get eight different combinations.

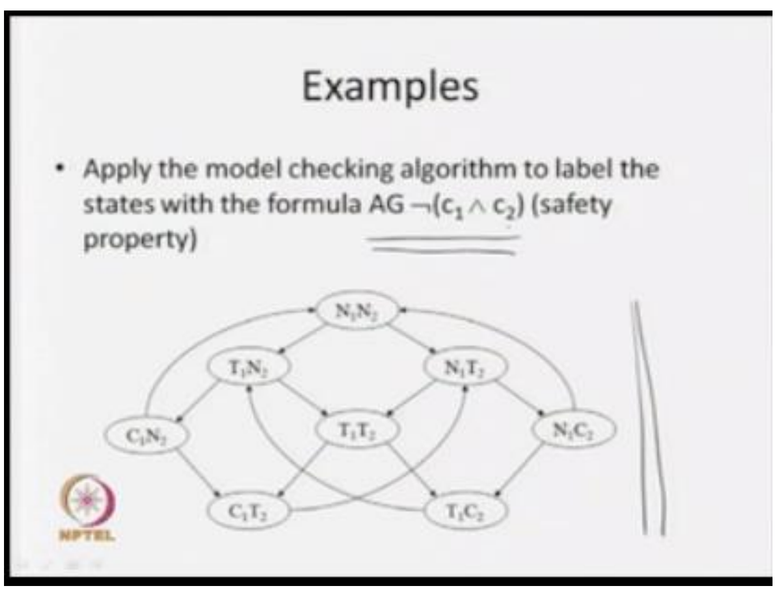
But you see the adequate set of operator so if you know the procedure for these three operators that we can look for the truth values of any other CTL operators. Now in last class we have seen the algorithm and these are these algorithms are some sort of your graph traversal algorithm, so we are going to take the kripke structure with your set of states all the transition and along with the labeling function.

Now in all this procedure what happens we are going to propose the entire state space and the graph and I am going to check whether the property is true in a particular state or not. So in complexity issue we have said that this depends on the length of the formula, so what is the length of the formula then it is, because in each and every state we have to look for it, this is your number of notes in this particular graph.

And along with that V+t so this is basically the total state space, because you have to traverse in all citizen its transition so the complexity this turn off to be your order of your length of the formula and number of notes and your total state space so it in linear in the length of the formula and what in taking the number of notes that we have in this process. And we have seen that we have got an well defined algorithm or this operator so that means we can say that we are having automated process for CTL model checking.

And this is the beauty of CTL model checking so we are getting an well defined algorithm and it transform linear time okay. Now we will see the huge of this algorithm with alpha example, so here in today's lecture I am going to discuss some examples only and if time permits and we will see some raises also.

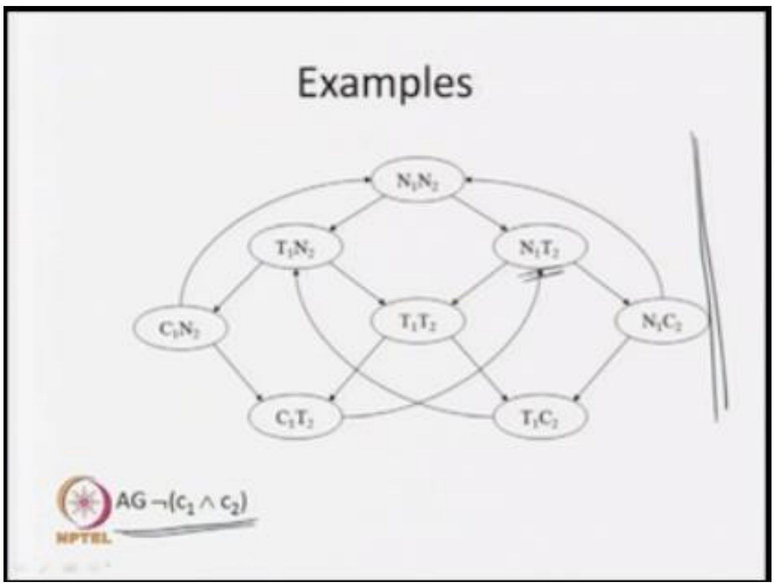
(Refer Slide Time: 04:14)



In one of our class we have discussed about the modeling if our mutual exclusion protocol and going to check for the properties that need to be satisfied that particular model okay. We have considered model of two processes P1 and P2 and we have seen the states that this run to up to be something like that, that we have presented over here. Already we have discussed this particular example in details.

So these are the reachable states that we have below our examples, now we are going to check whether the safety property satisfied or not in this particular model okay so what is the safety properties we are going to say that in all part globally ¬of C1 and C2 that means at any point of time both the process should not enter into the critical sections so both of C1 and C2 may not be true simultaneously okay so this is C1 is basically says that process P1 is your critical section and C2 say that process P2 is in your critical section so now property safety property in CTL is AG¬of C1 and c2 okay.

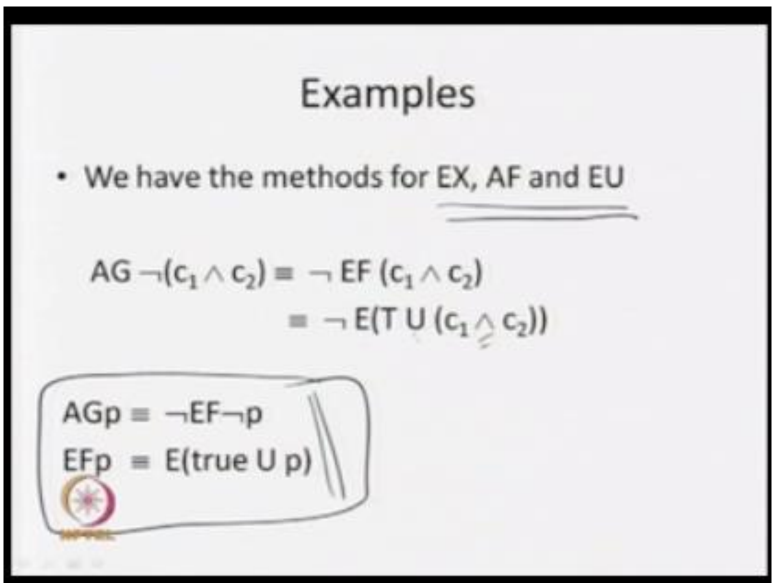
(Refer Slide Time: 05:25)



So basically this is the thing that now we are having you no hand this is the model or keep the structure that we have and this the formula that we are going to check weather what are in which are the state this formula is true okay so in sense it is a critical structure we are having the leveling of those particular atomic proposition in each and every step like that simple example I can say that N1 T1 it means that the process P1 is in none critical section and process P2 is lying into that time to enter into the critical section.

So that is why I am saying that the atomic proposition N1 and P2 are two in this particular states and all other states are this all other proposition like N2 P1 C1 and c2 are false in this particular state now if you say that this is a very small system and by inspecting itself we can say that in none of the state that¬of C1 and C2 is true okay but still if we are going for bigger system by inspecting we cannot do it we have to apply our algorithm so for this small example we will apply our algorithm now let us see what will happen.

(Refer Slide Time: 06:32)

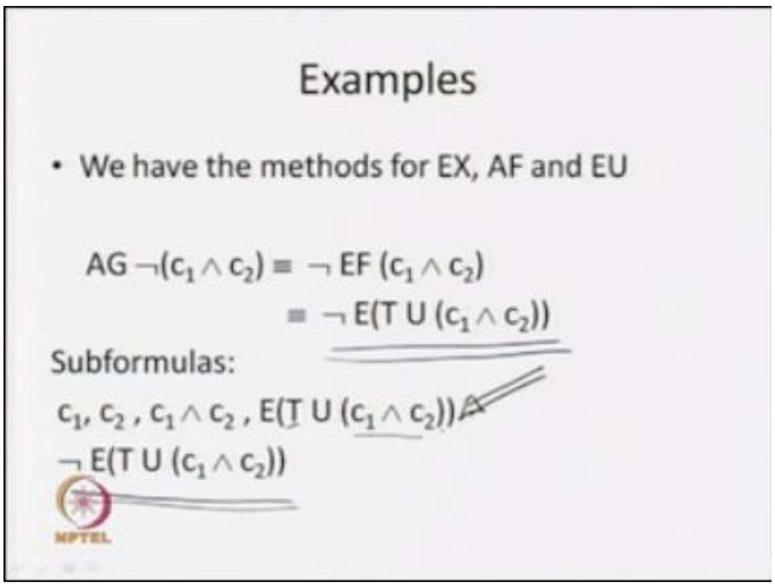


Now we have seen that we are going to look into the educate state of operators and we say that these are the 3 operators that you need EX AF and AU and we have already discuss the algorithm for these 3 operators so whatever CTL formula we are giving somehow we have to convert it to these 3 operators.

So already we have seen the equivalence about your CTL formulas and we now that AG can be expressed with the help of EX so AG¬ of C1 and C2 is equivalent to¬of EF C1 and C2 and this furthers can be again because if we have EF but we are having procedure for AF but we are having producer for EU so this EF can be written as E true until C1 and C2 because these are two equivalence that we have in hand so I have used these two equivalence AGP is equal to ¬ of EF ¬ P and EF Ø equal to E true until P so I have used these two equivalence and with the help of these equivalence I have converted this given CTL formula and you just say that we are having this particular conjunct we have this E until and we have this negation okay.

So we have the procedure for this E until so we are going to use this particular E until to check weather this property holds in some states or not.

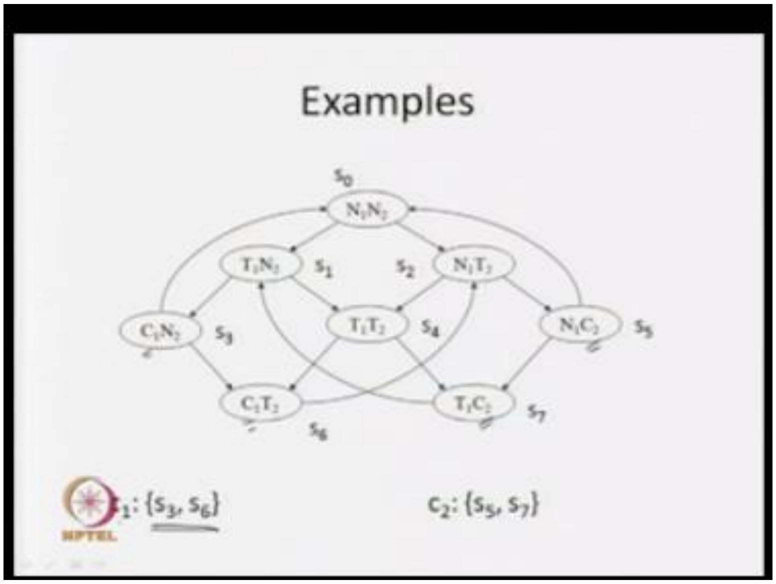
(Refer Slide Time: 08:00)



Now by looking into this particular final formula what are the sub formulas that we are having c1 and C2 are two atomic propositions so they are CTL formulas since C1 and C2 are CTL formula C1 and C2 is also a CTL formula now E true until C1 and C2 since C1 and C2 is a CTL formula true is also CTL formula so E true until C1 C2 is also a CTL formula since E true until c1 c2 is a state formula so negation of this formula is also CTL formula so this is the formula that we are going to look into it but to check for this particular formula we have to look for this particular sun formula once we have the leveling of this particular sun formulas then only we can talk about this particular given formula .

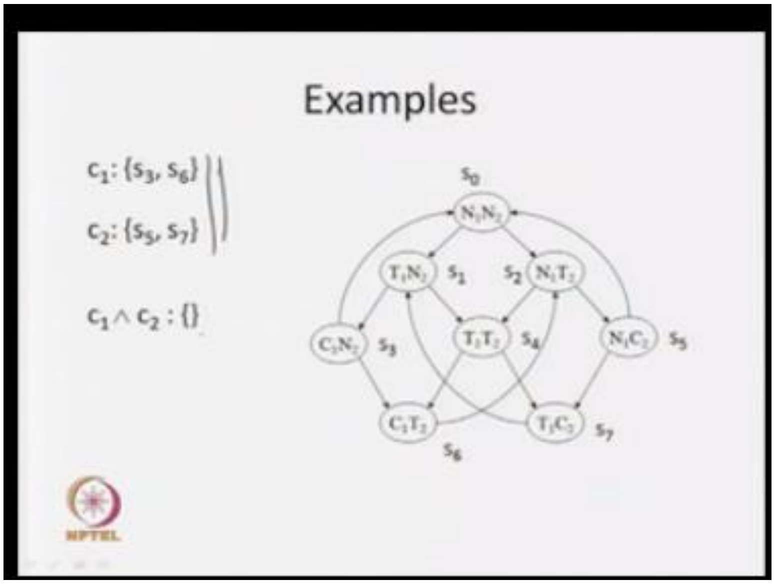
So we will look for the atomic once C1 C2 then C1 and C1 can this particular E true until C1 and C2.

(Refer Slide Time: 08:56)



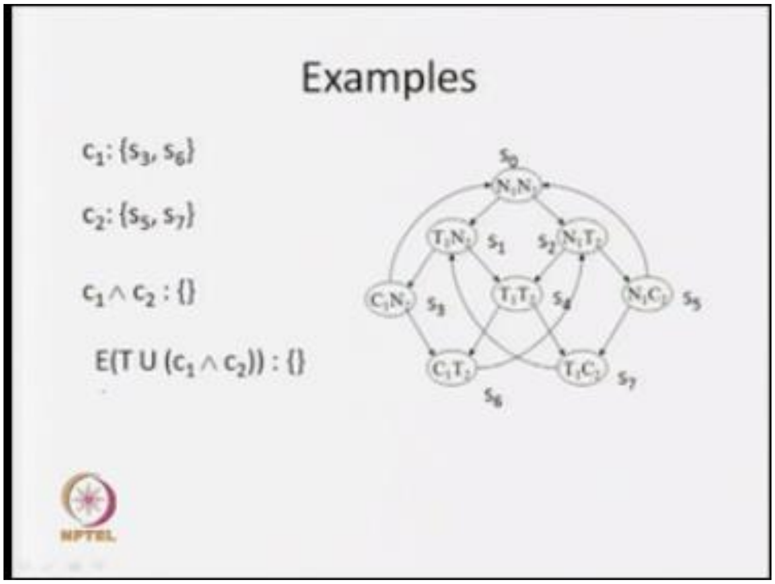
So now say we are having the leveling function we are level each and every sate with the help of atomic proposition so with the from this particular leveling function we will get that C1 that atomic proposition C1 is 2 in your state S3 and S6 so in S3 C1 is true and S6 C1 is true similarly for C2 it is true in S5 and S7 so this is the C2 it is S5 is level with C2 and, lower S7 is level with C2 so the SAT states where C2 is your S5 and S7 so we know the leveling of these two sub formulas C1 and C2 once we know that leveling of this C1 and C2.

(Refer Slide Time: 09:42)



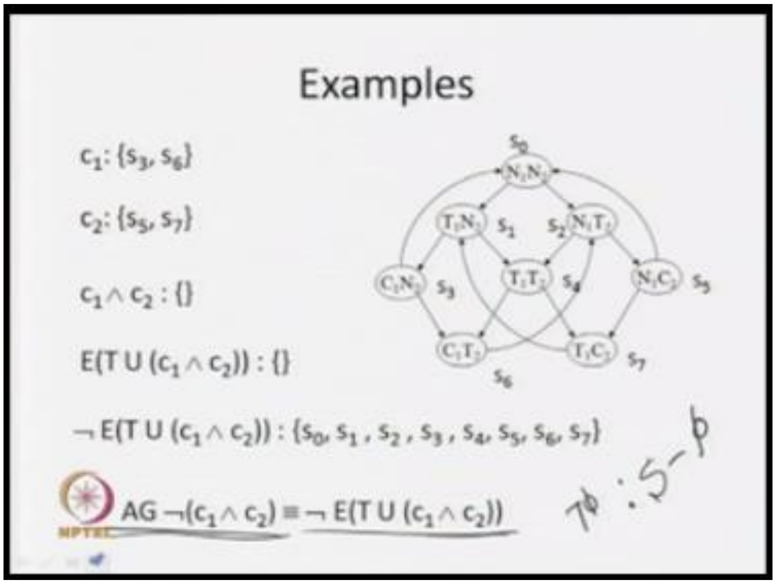
Then we can look for the leveling of C1 and C2 and what is the state were C1 and C2 is to this is nothing but the intersection of these two particular states if you look into the intersection of these two particular state then we are going to get a null state so in none of the states that C1 and C2 is true.

(Refer Slide Time: 10:02)



Now what is our next formula this is your E2 until C1 and C2 okay in this case also now it variable we are going to get the null set because C1 and C2 is null okay so in first case we are not getting any state so if we are getting some state then what will happen we are going to refers the graph since the initial state itself null so there is no point of refers in the graph so eventually we are going to that E T true until C1 and C2 is also null set okay.

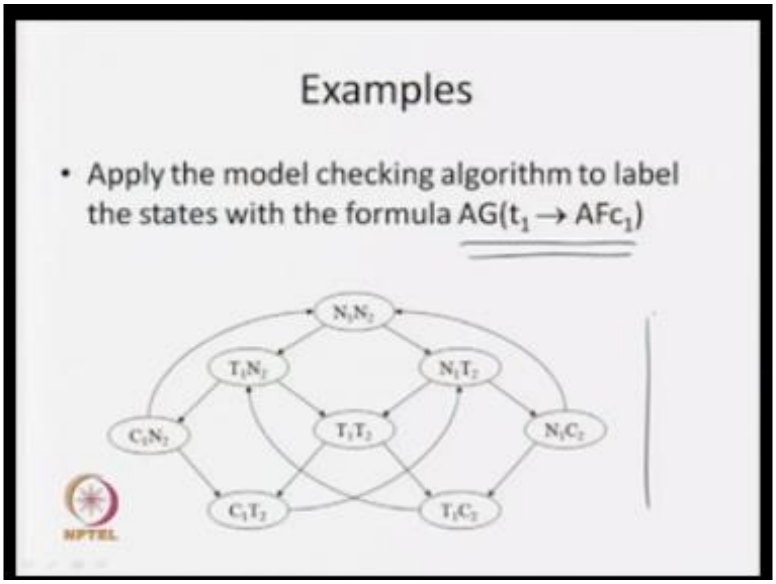
(Refer Slide Time: 10:35)



Now E T1 until T2 is a null set then what is the negation of these things negation of these things is negation of these things is nothing but complete state + - where the formula is Ø is true so this is basically these states give me the operator formula naught of Ø is true okay so total state space is given over here so my this particular the formula where this particular the states where this particular formula is true is given as your empty set or null set so for negation of this formula will be true in all the state okay.

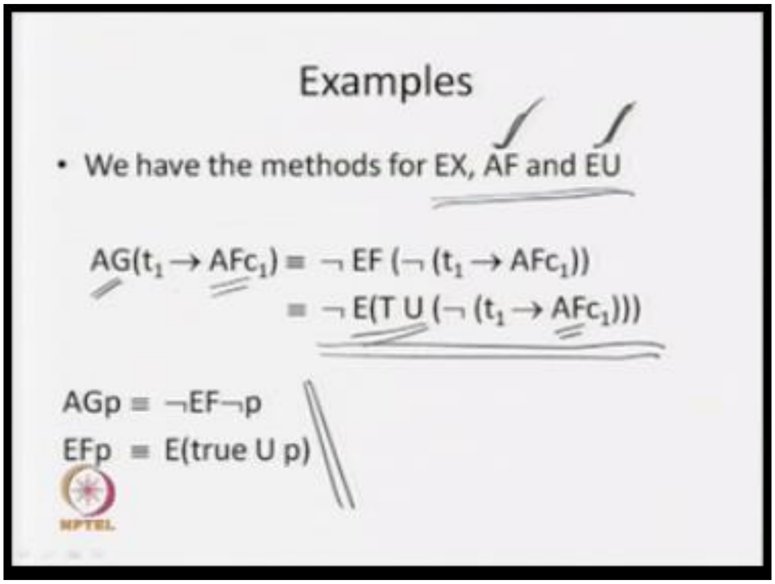
So that means what we are saying that this is the formula AG naught of C1 and C2 which is equivalent to naught of E 2 until C1 and C2 so you have seen that in all the state this naught top E to until C1 is C2 that means you can say that in all states all path globally naught of c1 and c2 is true so form we can conclude that the set 3 property as been satisfied by my model okay this is way that we can check. This is a very simple example now we will see one another example.

(Refer Slide Time: 11:44)



Okay in second example we are saying that again we are going to apply the model checking algorithm to label the states with the formula all path globally T1 → where in all path in further C1 okay so this is the property okay so this is the state property that we call as your life less property so what we are saying that if any possess once to enter in the critical section that process eventually must get the sums to enter into critical section this sis the life less property now we are going to check whether this particular life less property is through in my model or not okay.

(Refer Slide Time: 12:23)

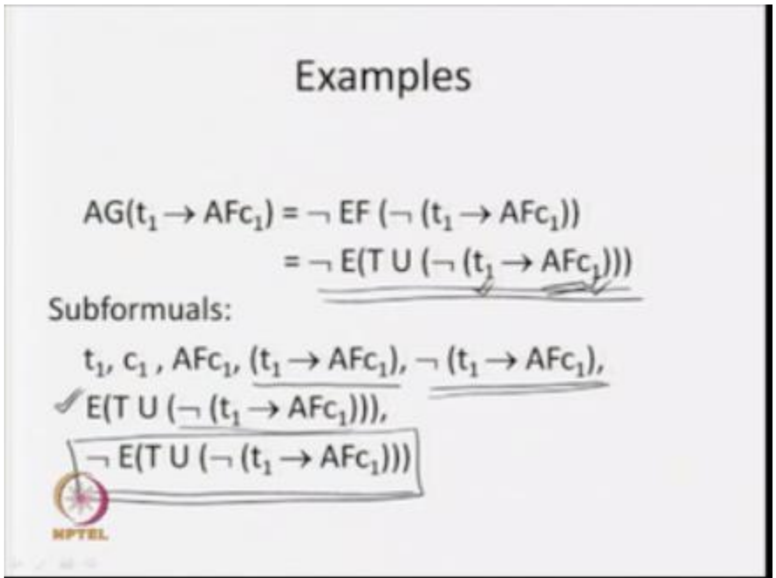


Now again you see that we are having method for EX, AF and EU so somehow we have to express this particular formula in terms of these three formulas okay now AF we are having so we have process or procedure to check for this particular formula AF C1 but we do not have method for particular AG so that is why we need to use some equivalent.

So that we are going you have method or process from this particular set now again similarly say AG p is equivalent to naught of EF naught of p this is equivalent and the second equivalent is EF p is = E true until p so we are again going to use these two equivalent so eventually this event formula AGt1 → AFc1 is turn up to be this particular formula naught of here exits 2 until naught of t1 → AF c1.

So this is the formula now we have process for AFc1 we have process for E until so these are the E until and this, the AF so we can use all model checking algorithm to check this particular formula.

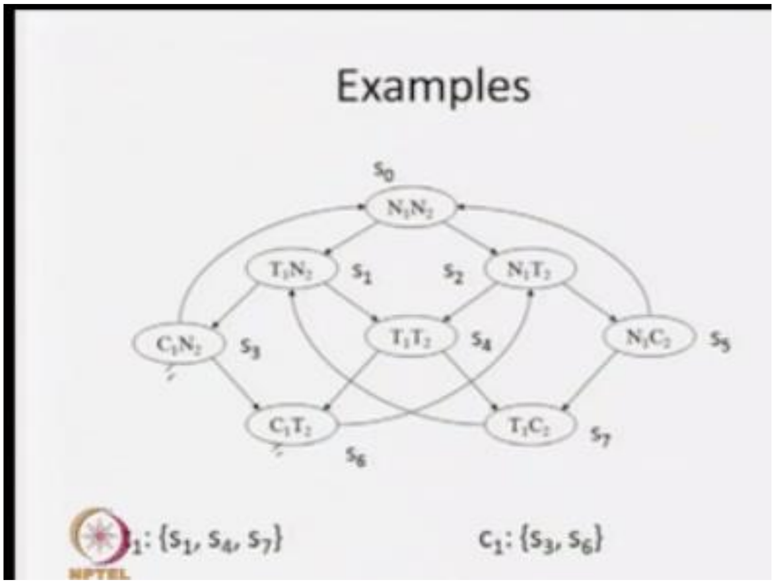
(Refer Slide Time: 13:41)



So now one we are converting this particular given formula AG t1 → AFc1 and this is the equivalent formula that we are getting and now we are concern about this particular formula now what are the sun formulas is this particular formula now I can destining say t1 is sun formula C1 is a sun formula since c1 is a sub formula then AF C1 is also a sub formula since T1 and AFC1 is also sub formula so T1 implies AFC1 will also be a sub formula or CTL Formula sub formula since this one is CTL Formula sub formula negation of this given CTL Formula is again a CTL Formula .

So we are going to get one sub formula so from that now since both the components two is always a CTL Formula 2U we are having this as a CTL Formula so we are going to consider this as another sub formula and this is the final formula negation of this particular step, so when we are going to look for the truth values of this particular given formula we must know that truth values of all those particular sub formula. Know the set of states where this sub formula are true or the states where this is satisfying okay.

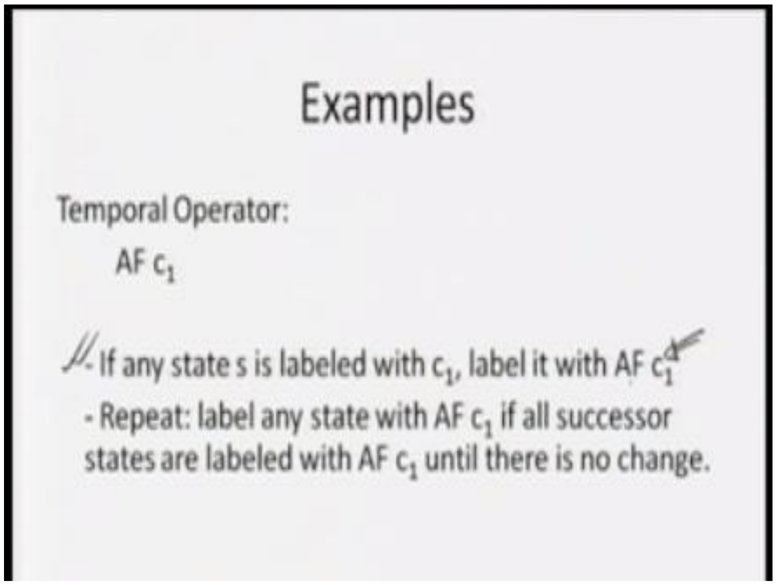
(Refer Slide Time: 14:55)



Now we are going to see these things now the sub formula we have T1 and C1 which are your basically atomic proposition so by looking into this particular state space we will say that T1 is true in the set S1, S4 and S7 so this is the state S1, S4 and S7 in this three particular states the atomic proposition T1 is true, similarly the set of states where C2 is true this is your S3 C1 is true in S3 and S6.

So this is S3 and S6 in these two states the atomic proposition C1 is true in non of the state C1 and C2 is true so we are getting these two states where formula T1 and C1 is true.

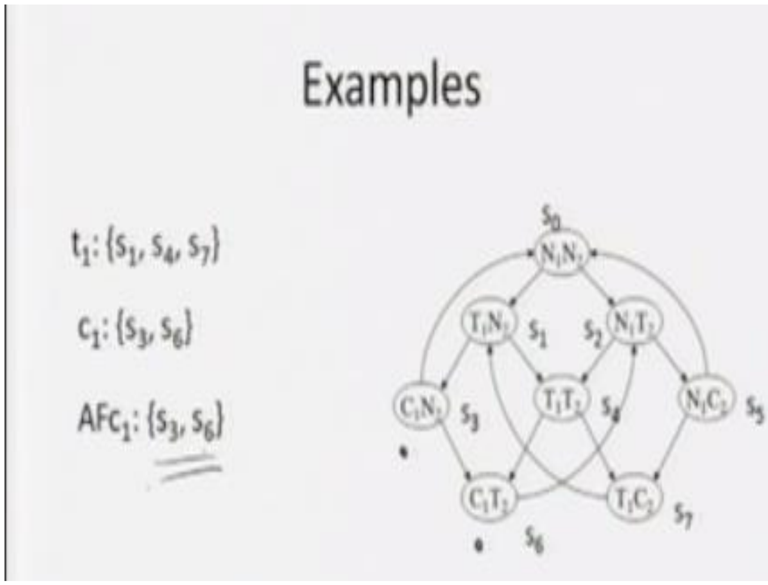
(Refer Slide Time: 15:49)



Now nest sub formula is your AFC1 we know the set of states where T1 and C1 is tier now we will look for AFC1. So what is the procedure for AFC1, so for any temporal operator follow this procedure we are having two step first step it says that if any state S is leveled with C1 can level it with AFC1 that means wherever C1 is true AFC1 is also true, because it is coming from what semantics of our future.

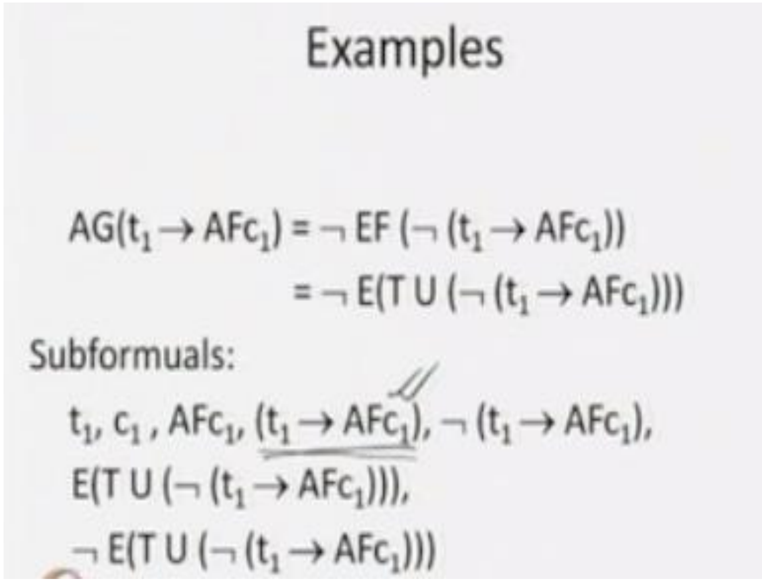
Because these are encrypt the present scenario, after that we are going to repeat this process level any state with AFC1 if all successors state are leveled with AFC1 until there is no change sop we are going to repeat this things now once wherever C1 is true we are going to say that AFC1 is true and we will repeat it.

(Refer Slide Time: 16:30)



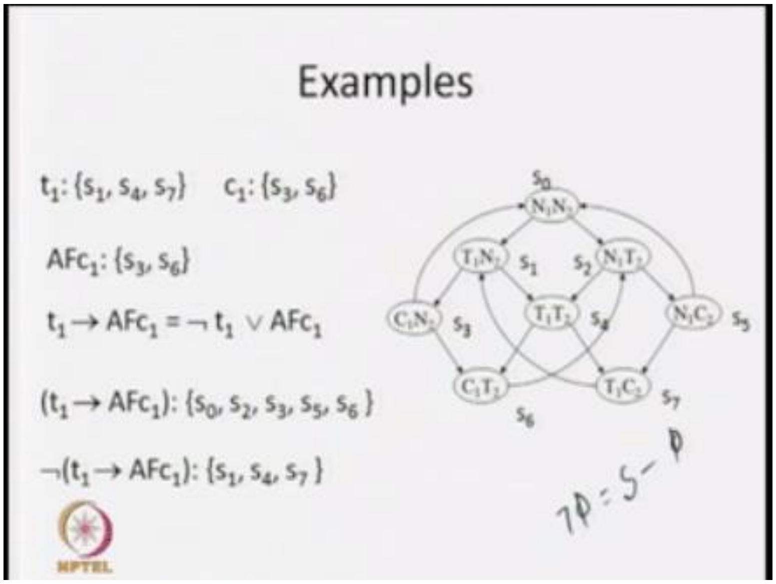
Now since C1 is true in your S3 and S6 so AFC1 is also true S3 + S6 now this is the base case that we are having now we are going to enter into the loop now from this two state we are going to traverse the craft in backward direction we will see what is the predecessor of this particular states okay now since the predecessor is your S1 now we will see where the it is having two next state S3 and S4 now S3 is leveled with your AFC1 in our best case but your S4 is not leveled with C1 so S1 both the successor are not leveled with your AFC1 so that one S1 will not include in this particular case since S1 is not include over there is so there is no chance of this particular state so we will staminate that one that means AFC1 is true in this true step S3 and S6.

(Refer Slide Time: 17:35)



Now the next formula we are getting T1 implies AFC1 now when we are coming to this particular sub formula we know the leveling of these three sub formula T1, C1 and AFC1.

(Refer Slide Time: 17:49)

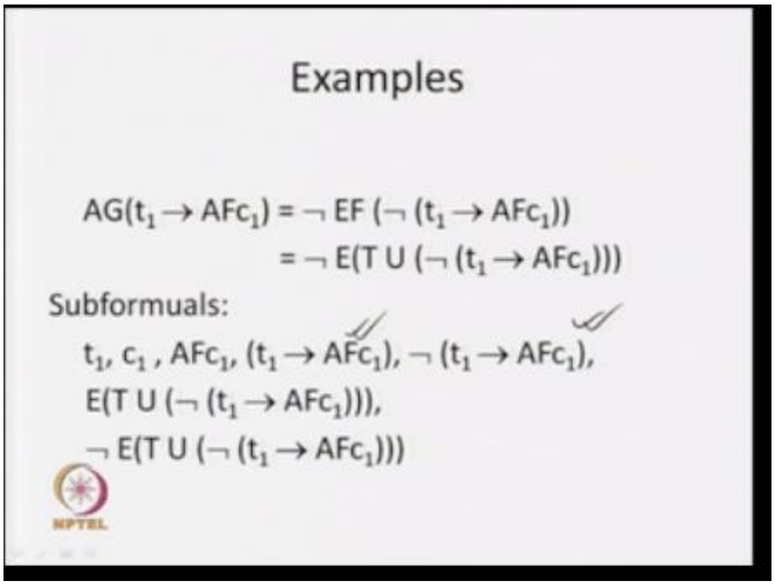


Now I can in this particular say that T1 implies AFC1 is equivalent to not of T1 or AFC1, so this is simple your logical connective so in other step we can check it we need not to go for a loop so now we are going to check where T1 is false we know that T1 is true in your S1, S4 and S7 so not T1 will be true in the remaining states and C1 is three in your S3 and S6 since it is true in S3 and S6 concisely these two states are going to come.

So T1 implies AFC1 it with true in your S0, S2, S3 because S0, S2, S3 S0, S2 and S5 in this particular three state not T1 is true okay, not of T1 is so due to this component these three states are coming and AFC1 is true in other two state your s3 and S6 so these are coming due to this on other hand not of T1 is also true over here so we are going to get these thing but the remaining three state S1, S4 and S7 in this particular three state AFC1 is not true and similarly not of T1 is also not true.

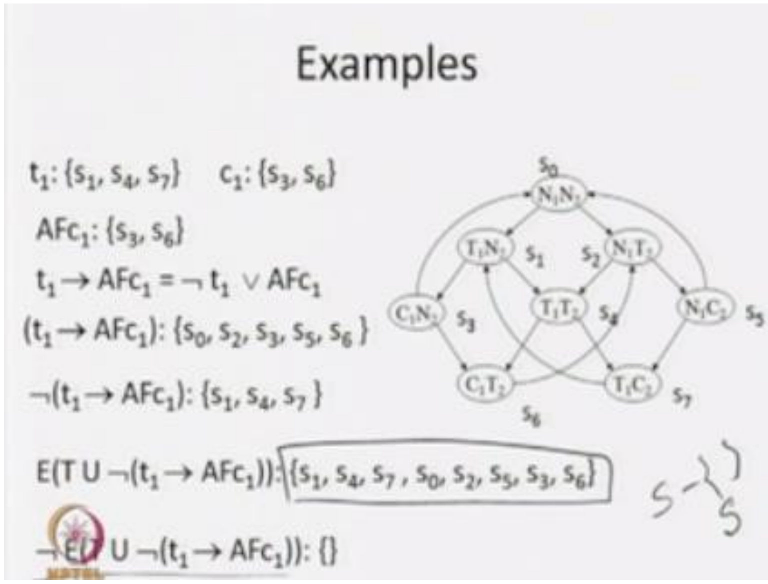
Because T1 is true in this particular state so these three states are not coming into so T1 implies AFC1s this basically true in s0, s2, s3, s5 and s6, so what is not of these things already I have send that if phi if true in some state then not of phi will be true in the total state space minus the state of set where phi is true, since this particular formula is true in S0, S2, S3, S5 and S6 so not T1 implies AFC1 will be true in s1, s4, s7 these are the three state S1, S4, S7.

(Refer Slide Time: 20:00)



Okay now we have to see the leveling of this particular sub formula along with that negation of is particular sub formula, so what is my next sub formula, that E true until not T1 implies AFC1 okay now we are going to check for this particular thing.

(Refer Slide Time: 20:19)



So what is the procedure for EU operator so if any state S is leveled with Q then leveled it with EP U q again s power semantic these are the increase the present behavior okay so wherever Q is true we are going to level it with EP U Q the we are going to repeat this particular procedure until there is no sense what is the procedure level any state with EP U Q if it is leveled with P and at least one or with accessories leveled with E P U Q.

So first we are going to collect the set where Q is true with level these things with EP U Q in then we are going to transverse the entire states space and we are going to check if the P is true in this particular state and one of these accessories leveled with EP U Q and until there is no sense that means we cannot collect any more states, if you cannot collect any more state then we are going to terminate or produce your over here, okay.

Now in this particular case say we know not T1 implies AFC1 okay we know these are the what are the step it is true in S1, S4 and S7 for each true until not T1 implies AFC1 this will be true in your s1, s4 and s7, s4 for our for step in values that s is leveled with Q then leveled with it E P U

Q so from this particular step what we are getting that this particular formula will be true in S1, S4 and S7 because this not T1 implies AFC1 is true in S1 , S4 and S7.

So these are the initial step we are collecting now from this particular three state we are going to traverse the state space okay now next thin what will happen now when we are going to look into your S1 then we will see one or which this is as your S0, okay. Now S0 is leveled with Q no we have seen that it is level because all the states are leveled with Q so true until something so if it is leveled with true.

And one of these accessories leveled with this particular your AE U your distinct because already we have got these are the three states so now we are going to include this particular S0 in this okay so it is not having any other procedure then we will go to S4 from S4 again we will see what are the procedure it is having one producer and these producer is also leveled with true and now since true it is leveled with true and one of this accessories leveled with already your this particular sub formula so S2 will also come into this particular state, similarly and now it is not having any other procedure.

Then we will go to S7 now from S7 we are going to see what are the procedure it is having of course from S4 we are having S1 also but it already leveled so from S7 we are having 226 S5 and S4 so S4 is already leveled now in S5 we will see that it is already leveled with true and there exist that is one successor which is leveled with your this particular given formula AFC your these things.

Not of T1 U EFC1 that means it is leveled with this particular given formula so S5 will also come into this particular thing, so in this situation we have seen the producer of all these three states and we are introducing so these thing we are getting a big state so there is a sense of my set of states so again we are going to repeat this particular loop so next time what will happen, now already we have seen these three things.

Now we are going to see S0 now when we are having SS0 then you just see that what is the predecessor that we have so we have this particular predecessor because so we are having a transition formula S3 and we are having a transition formula S5 so S5 is already leveled okay now we are going to see that S3 now I can as far our definition it is leveled with 3, 2 and it is having two successor but one of the successor is leveled with your that particular formula.

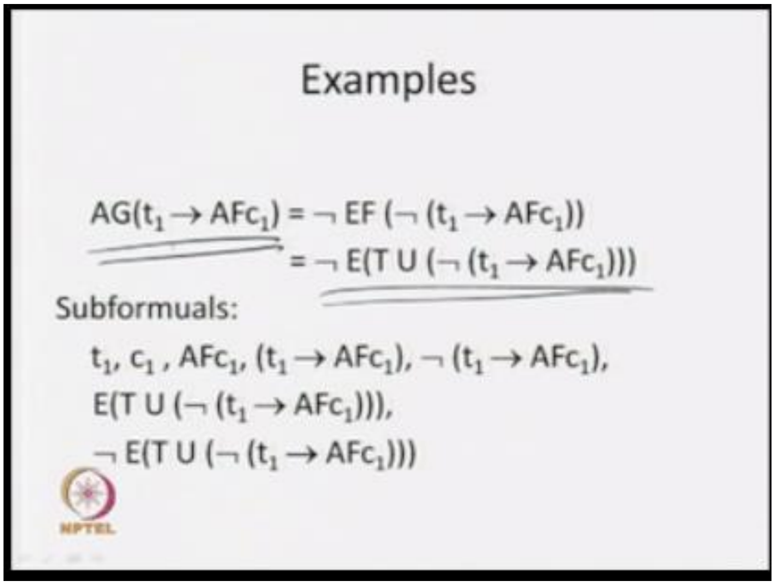
Because already the pervious step we have leveled with so you are S0 will come into this particular state okay, so S0 sorry S0 is already included so SS3 will also be included in this particular state, okay. Now after that already S5 is there so we are already have included it so now we have seen that again there is a change in this particular state so in again and trying to allow so when we enter into the next loop.

Then what will happen, already we have checked those particular state now only S3 is reaming so from S3 we will see that what is this particular successor S6 now I can in S6 it is leveled with true now one after these things we are having these things what we call it is your S3 is leveled with you are the given formula E2 U not of T1 implies AFC1 so S6 will be included in this particular state and the second processor that we are having so if in a two look into it.

So you consider that since during the sense again we will go into it now we will find that when were enter into the next step then all are included over here because we have seen so all the states are we included so there will be no sense so we are going to get in all those particular state that means entire states space E2 U not T2 implies AFC1 is true, now my given formula is negation of this things.

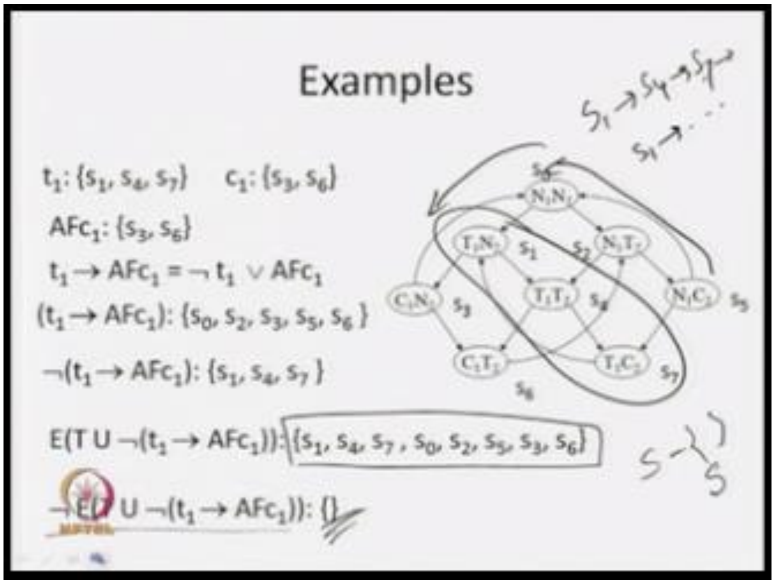
Since in we are going to have S minus this particular state so since this is nothing but equal to your state so we going to get a non state that means the given formula is not true in non of this particular states, okay.

(Refer Slide Time: 26:53)



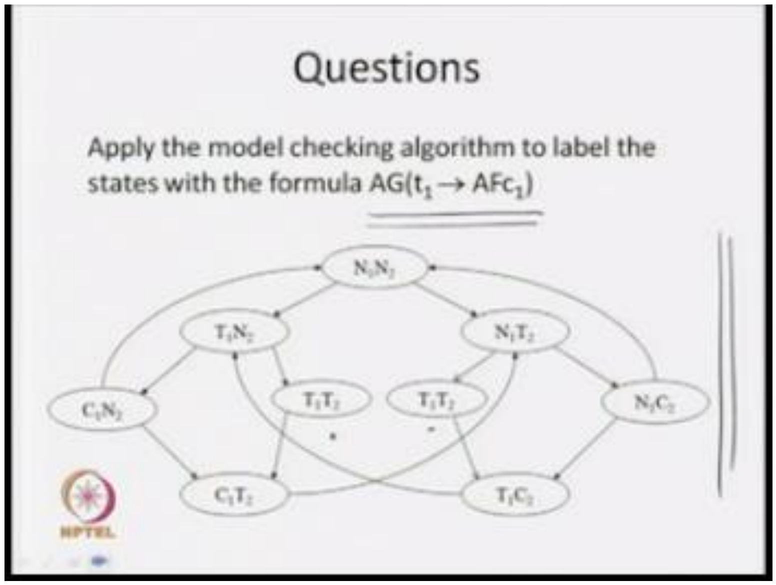
So this is the given how we have looked for the truth values of this particular formula and this is equivalent through my given formula and we have find that in none of the states these particular formula is true that means our modeling of our distance what will happen the model that we have no it is not satisfying all life less property so if any process trys to enter into the critical section eventually it may not get a some enter into the critical section okay because in non of the state this particular event formula is true.

(Refer Slide Time: 27:27)



So all ready we have explained that this is happening because of these particular loop okay so we are coming from S1 to S4 from S4 to S7 and from S7 to again we are going back to S1 like that so we are going to get an infinite path in this particular loop so from any sates if you look into other states for every states we are going to enter into this particular loop so if for example say if I am in S5 from S5 I can come to S0 from S0 I can come o S1 and after once this into the s1 then s1,s4, s7 it will remain in this particular loop, so due to this particular loop this live ness property is not prove in our model, okay. So already we have explained these things now we try to apply our model sign algorithm leveling algorithm eventually found it that it is retaining with the null step that means in none of the step the given formula is true, okay.

(Refer Slide Time: 28:27)

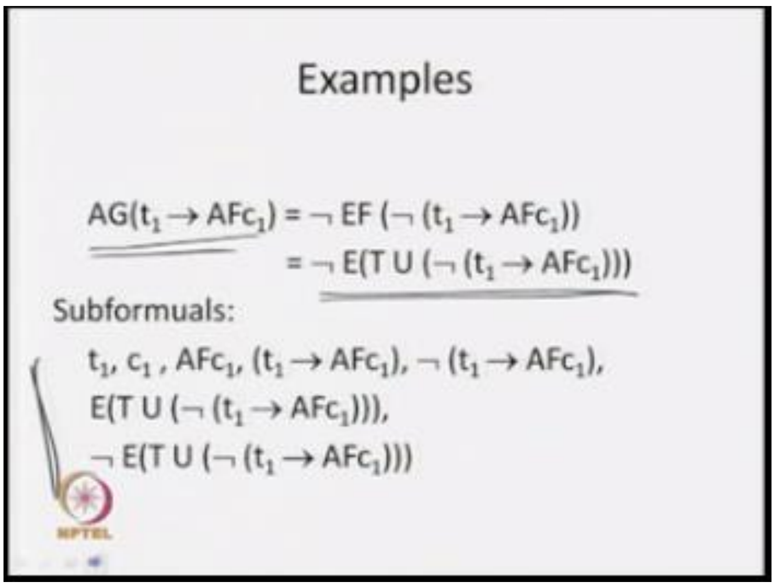


Now after that save when I said that we are coming up a model we are trying to say check for the property whether my property is true in this model or not we have found that the liveness property is not true, okay in the particular case what will happen designer is going to revisit the design he will analysis the design and he will try to modify the design in such a way that the required property will be satisfied in the new model.

So already we have discussed this in pervious the XR and we have seen that we can come up with a new model. Like that where it one states will be broken into two different states over here, okay. Now I am giving you question that apply the model checking algorithm to label the states with the formula AG t1 and the AFc1 so this is the live ness property already we have said in my previous model, now we apply the same procedure the way I have explain it and see try to check whether this property is true in this particular model or not, so that means you have to find out the set of states why this particular formula is true.

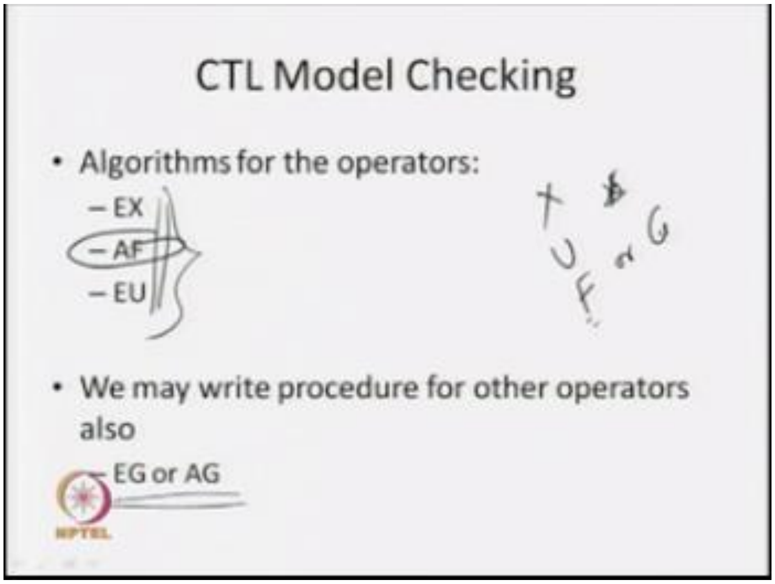
So it is similar to my previous explanation only this particular model is different we had one simple states over here now I have broken this particular state to two different state. Now with this approach now you try to look form try to get the equivalent same equivalent will come same sub formulas will come here also because I am giving you the same formulas only this model is different now we apply the model saying algorithm to label the states with this particular formula, this is a very simple example simple problem so you can apply it and thus see get a feeling how the models using the algorithm works over here.

(Refer Slide Time: 30:13)



So this is basically same problem I have already explain that this given formula can be break up into this particular equivalent formula and these are the sub formulas that we are having now you look for the set of state where those particular sub formulas are doing, so it is similar to the example that I have explain but here what will happen I have slightly since the model so at this property it will satisfied, okay you take it is an exercise and do it.

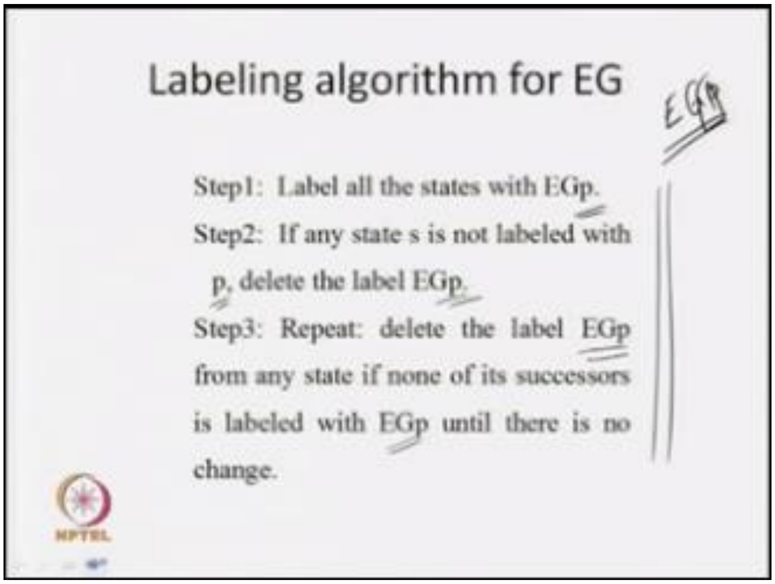
(Refer Slide Time: 30:44)



Now what will happen, we have seen that minimum set of operators we are going to look for EX, AF and EU and other operators can be explain with the help of these three operators. Now we will see some method that we can write some procedure for other operators also, other operators also or not. So for that I am going to take say some example of EG and AG because if I having EG or AG we have to write an equivalent formula which involve these three operator.

But we will see whether we can write procedure for these two or not, if we can write the procedure for these two then what will happen my adequate set of operator maybe different because already I have said that I need one next operator we need one until operator and I need either F or G so for F I have looking into AF okay, now for G also now we say that instead of F I can use G so I am going to see the procedure for EG and AG, okay.

(Refer Slide Time: 31:55)

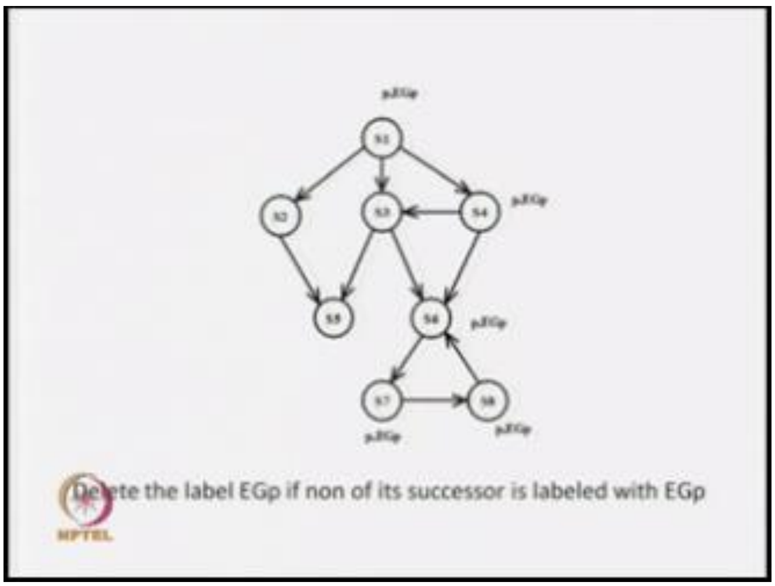


So first we are going to look into EG, now what we are going to see that leveling algorithm for EG so it is slightly define I am going to get it we will see that what is the first step I am going to do level all the states with EGp because I am going to look for the formula EGp, so when I am coming to this particular formula we know the leveling of this particular sub formula p, so in this particular sun formula we are going to do initially we are going to level all the state with your to a given formula EGp, so if now step 2 what we are going to do if any step s is not level with p we need the level EGp so what we are stating I am giving a step first in all the states I put the level EGp, okay.

Now after that I am going to check the steps where it is labeled not labeled with p, because we know the level of p know, when I am coming to EGp, so I am going to remove the label of EGp from those particular steps where it is not labeled p, okay. Now in step 3 now what we are going to do delete the label EGp from any state okay, if none of which successor is labeled with EGp, okay, so and I am going to repeat this process until there is no change that means I am going to remove the level and after removing the level what will happen some more steps will come whether that EGp is not true.

And like that we will repeat this particular process until there is no change that means we cannot remove any more levels, so this is the procedure that we are going to follow so this is slightly define from a earlier approach this is slightly define able, but still it is going to work and I will see what is differences, okay. Now these are the three step we have to follow and let us see what it is going to work or not and with a simple example I am going to explain it.

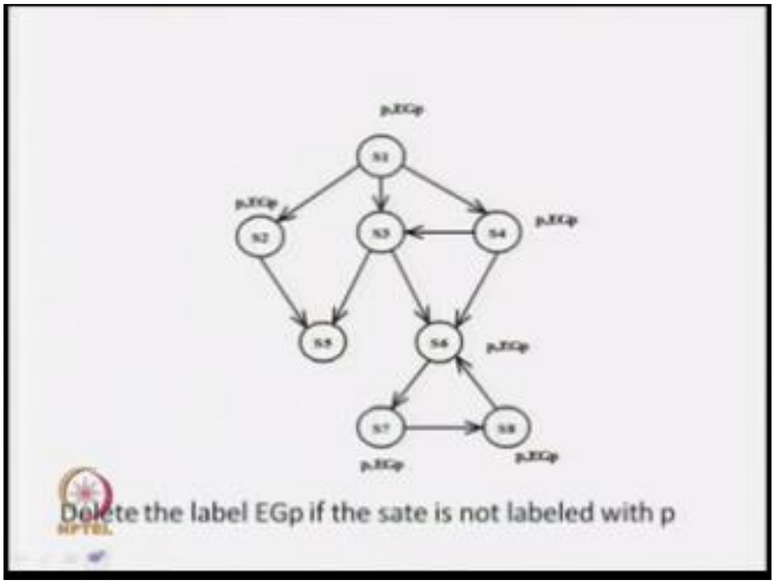
(Refer Slide Time: 34:03)



Now come up with this particular if a structure or model it is having the steps where s1,s2,s3,s4,s5,s6,s7,s8 and initially I am having the level of p when I am going for EGp I must have the level of p so these are the steps where it is labeled with p so basically I can say that p is true in the step s1,s2,s6,s7 and s8 okay, so in the first step level is set by EGp so I am leveling each step with the help of this particular EGp.

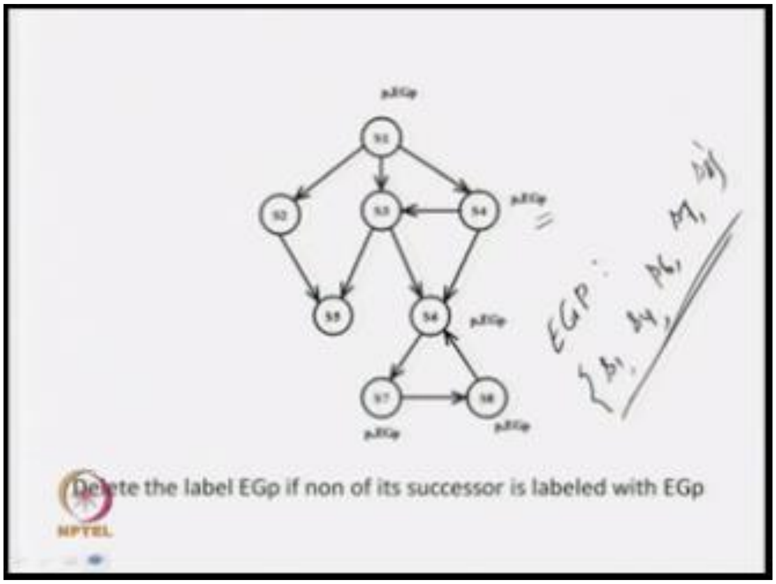
Now what is the next step, next step which says this remove the level of EGp from the steps where it is not labeled with EGp, so I have seen that it is p is level in this particular steps and it is not labeled in s5 and s6 so s5 and s6 is not labeled with your p, so I am removing the level EGp from these two step s3 and s5 so this is the step, now what we are going to do next I am going to do going to remove the level from the steps remove the level EGp from the steps when none of its successor is labeled with EGp. So in this particular case we will find that.

(Refer Slide Time: 35:39)



Already in the previous case I have removed the level from s3 and s5 now if you look into this particular state space if I come for your say s6 will find that at least 1 successor is labeled with EGp when you come s8 we will find that 1 successor is labeled with EGp, when you come s6 it is labeled one if the successor from e4 also e6 is level s1 also we are getting one, it is having two successor s3 and s4, so one of the successor is labeled of EGp but when you come to s2 it having only one successor s5 which is not labeled with EGp. So that means now we are going to remove this EGp from s2, hope there.

(Refer Slide Time: 36:21)

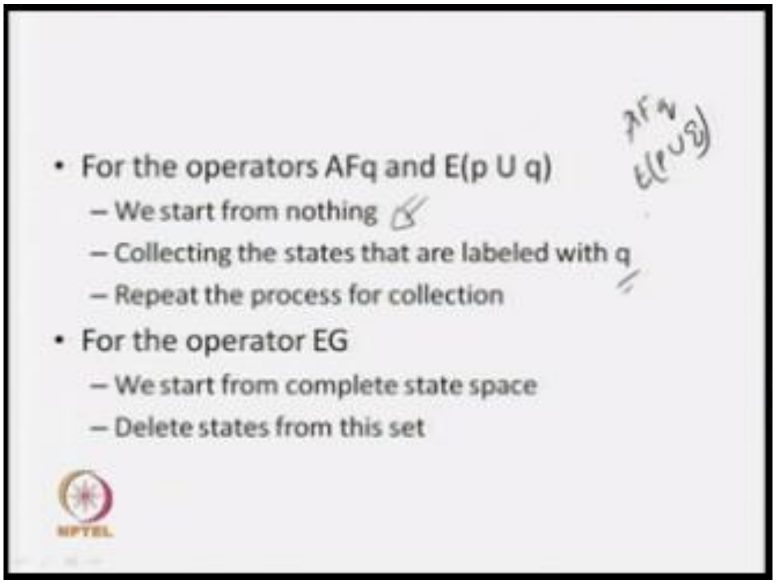


So you are removing now delete the level EGp if none of its labeled with EGp, so and we go take again I am going repeat these things so we will find that since we are removing it now this we have potential candidate s1 okay, now in this particular s1 what will happen you just see that from s3 we have removed it but it is having one successor s4 where it is labeled still labeled with EGp that means we cannot remove s1 so similarly we cannot remove the label EGp from any of the steps, so eventually our algorithm stops over here, okay.

So eventually what we are getting that set of state where EGp is true so we are getting the states like that s1,s4,s6,s7 and s8 these are the states because you just see that here from s1 at least they are exit a part where globally it will work. so similar for s4 also because this sub part from this particular s1 so again it will happen here also we will see that which we are getting states one part where p is true in all the states, so this is the set of states that we are getting over here, okay.

Now see we have seen the process for your EGp and the approaches slightly different then the procedure that we have talked about EU and AF okay, we have discuss the algorithm for EU AF and now we have seen the procedure for EG also, the approach of EG is slightly different in the approach of EU and AF. Now what is this difference?

(Refer Slide Time: 38:22)



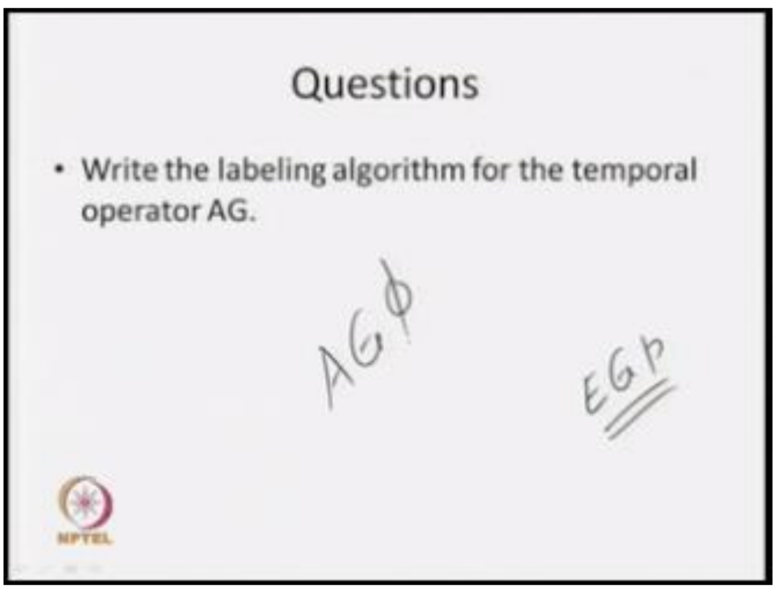
You just see that for operator AF and EU basically we start from nothing so we are not considering any states or we can say that initially we are starting from and this state. Now we are collecting the states data label with q so in AFq and E p until q so we know that if a particular state q is true then AFq is true and in this particular state if q is true then E p until q is true, okay. So we are collecting those particular states where q is true that means we are starting with minimal states this is the minimal state set of states where this particular formula AFq and EFq is true.

And after that we are going to repeat out process that means while repeating the process we are transferring the entire states space and we are collecting more and more states, okay we are adding more and more states and until we are going to repeat this particular procedure till one where we cannot add any more states, okay. So this is the procedure that means we are starting from a minimal state and we are trying to collect more and more states and eventually we will stop at the point where we cannot collect any more state, okay.

So this is the approach that we have used in case of AF and EG but in case of AF and your EQ EU okay, they exist the part until lower, but for the operator EG that means there exist part globally something is true or not so I say globally p is true or not. In this particular case our approach is like that we are starting from the enter states space we are initially thinking that in enter state space this particular formula is true after that what will happen we are trying to remove states from this particular state space by looking into the criteria whether EGp is true or what they are not.

So this is just you just see that approaches slightly different we are trying starting from the enter state space and now eliminating states where EGp is not true so slightly approaches different one is we are trying starting for the minimal state and in the second approach in EGp we are starting for the entire state space and trying to minimize it I am going to get the set of states whether EGp is true, okay.

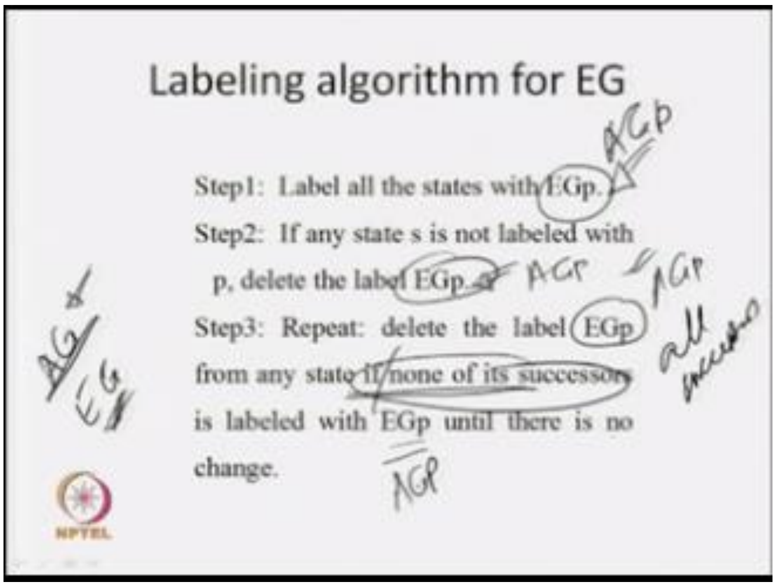
(Refer Slide Time: 40:47)



Now after discussing this particular EGp there exist but globally p holds or not, now I am giving a question to you because already we have seen this particular algorithm write the leveling algorithm for the temporal operator AG that means in all part globally some formula ϕ whole sum I am saying that write the leveling algorithm for the temporal operator AG part, okay. Now already we have discussed about EGp now I am saying that true or write the algorithm for this, I think this is a simple one.

If you look into it what is the next of the EGp and AGp straight away I think you can write a procedure for AGϕ okay, it is very well similar to EG, what we did in AG.

(Refer Slide Time: 41:38)



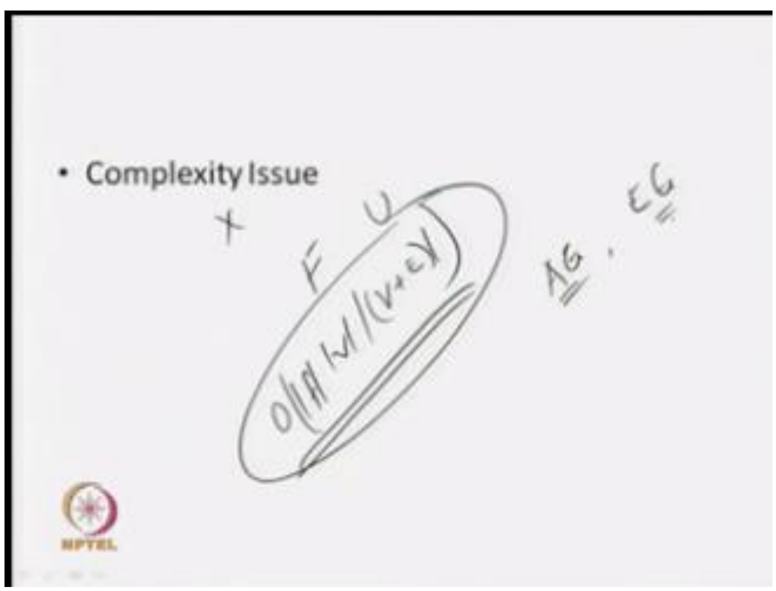
You just see that we are starting with the entire state space label all the states with your EGpm, now if any state s is not labeled with p then delete EGp okay, fine because you just see that AG and EG AG is a stronger condition than EG, EG may be true in some state, but AG may be false but if AG is true in some step then EG must be true over here because we are talking about all part EG we are talking about some part is from any part.

So if AG true in particular model then EG must be true but if it is true AG must be true okay so in this particular case we can say that if p is not level then we are deviating EGP is since EGP is not true that means AGP there is no question of having AGP over here.

So straighter e cans use this now what we are rippling over here delete the level EGP form any step if none of each accessories level with EGP okay what we are saying that if non of each successor is level so we are considering it is in case of you EG it none of it accessories level with EGP now for AG I am already telling you which is the stronger condition so instead of none of which successor what you say for any step if none of the successor is level with AGP we have to say all success of this.

We have to look for all successor because it is your AG in all part globally and it is stronger condition when it is so you just see that straight a way you can say come for your AG instead of none of the successor you can say that all successor or not leveled with your AGP and remove then re move this level until dismiss so level it state with your AGP if any of s is not level with p delete AGP form this particular step and delete the level AGP form any step[if all successor are not level with AGP until there is no step. You just say that since EG is the stronger condition so from EG we can carry on to this.

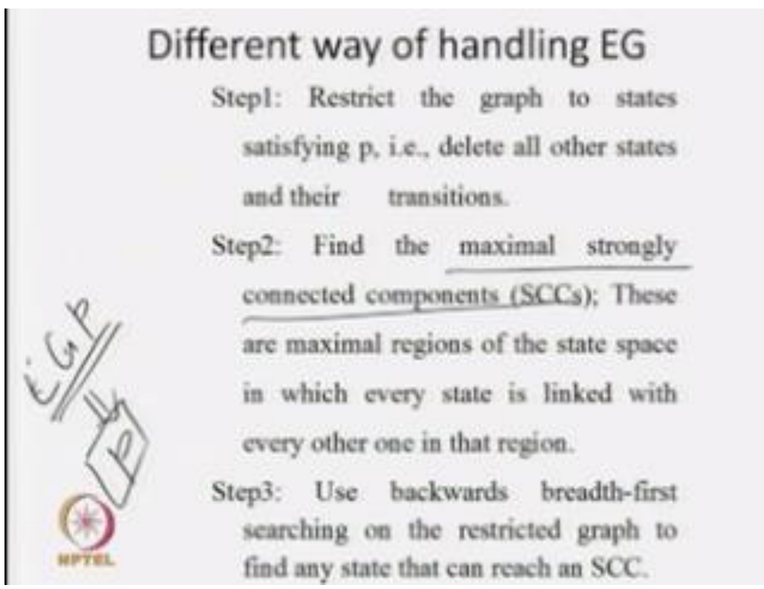
(Refer Slide Time: 44:08)



Now what are the complexity issues of this particular AG or EG already I have said that we have discuss about your in future F ᴗ and of course in both the cases we need this particular next step so this is basically complex we are talked about that it depends on lent of formula number of state we having and the total step states okay, so this now we have seen this is ion case of f and now if you look in to AG and EG now we are starting from the total state space but again we have to travel our enter travel space and we have to look for each step whether any one of this successor all successor all level with GEP or not.

So complexity will remains then for AG and EG also but whether can we improve this particular complexity, sop can you do in some other way so if you look for AG and EG we may have some other way for this two operator just will see one of this particular case.

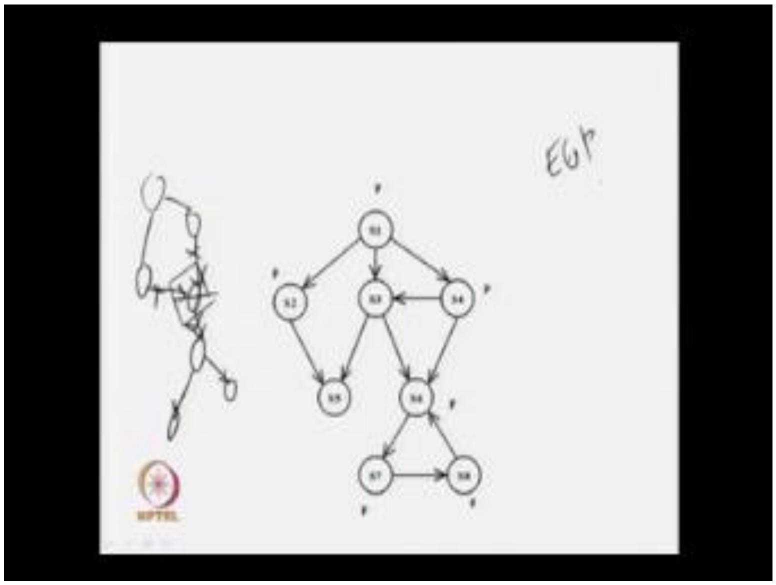
(Refer Slide Time: 45:25)



Now what is the way that we are going to with now this is slightly defined a pros now see the algorithm in step one we are saying that restrict the graph to states satisfying p that is delete all are the states and their transition, now what we are saying we are giving with the ultra state space when we are going to look for EGP that means what will happen when I come to EGP that means I know that step one is already level by this particular formula p or it is my proposition.

Now we will take the ultra step space and I am going to walk with the restrict at top what we are going to say that delete the steps were p is not true and along with this transition so we are restating it.

(Refer Slide Time: 46:17)



So basically we consider going to look at with step if I am saying a state space like that okay just it mean now say this particular state is not have a level of p that means we have going to remove this particular steps then when I am removing this particular step okay done what will happen is transition will remain daggling so we have to remove this particular transition also okay. So initially we are starting with this thing so we are going to restrict across by deleting the state where p is not true okay.

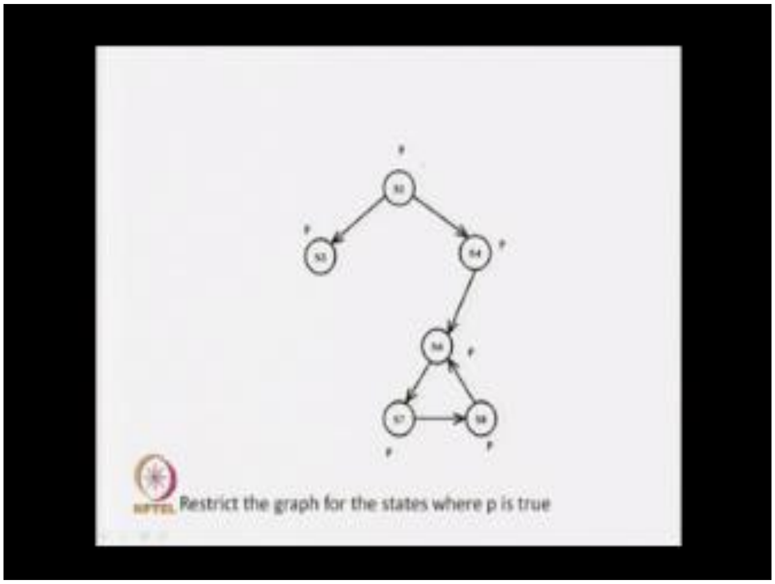
And along with your transition so we are going to restrict it up with the states were p is true now next step in stake to what we are going to find the maximal strongly connected component of this state, so this is maximal strongly connected component that we are going to get we having the procedure to do this thing so what is the maximal connected component these are maximal regions of the state space in which every state is link with every aversion.

Basically maximal strongly connected component basically these are the complete graph complete soft craft of my third restrict dept come so that is from any state we can go any other step of course it is indent to complete that we are having a part from any state to any other states okay. So this is maximal strongly connected component indent to complete okay. Now in step three what we are going to do use backward breadth – first searching on the restricted graph to find any state that reach an SCC.

Now we going to follow the backward breadth – first search on this particular restrict of to find any step that can reach an SCC, so we are identifying the SCC strongly connected component okay now from this particular strongly connected component we use backward breadth- first search okay I am going to shift whether from any state reach this particular state all, so if we conveys that informs the particular state on ward it will be true okay.

Now you see come to this particular same example that this is the state first that we are having we are having state from S1 to Sa and we are going for EGP okay so in that particular case what will happen it is already the step space is level with this state, so these are the state were p is true. Now what happens it is already the states are leveled with this so these are the space where P is true.

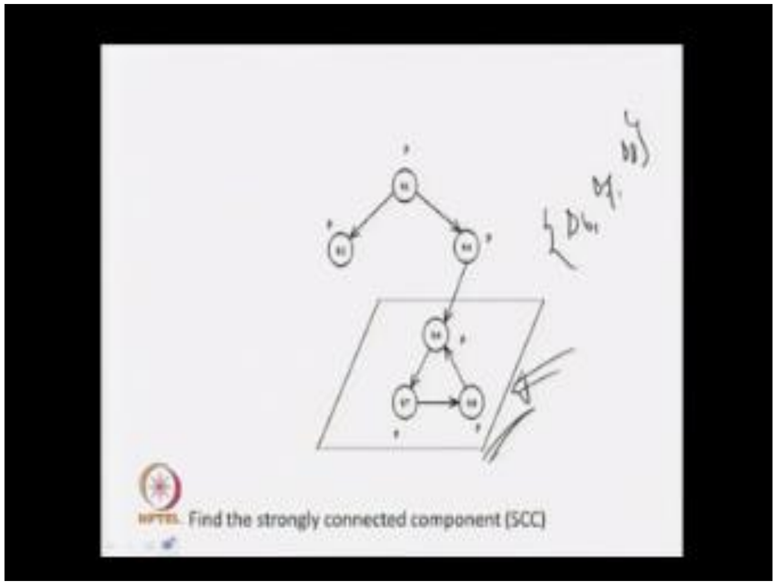
(Refer Slide Time: 49:04)



Now what happens a restrict graph for the states where P is true. In this particular case if you see out of those steps S5 and S3 are not leveled with P. So we are going to restrict our graph where P is true that means we are going to delete these two particular steps when we are going to delete these two steps then along with that you have to delete all those particular condition. Which are coming in those particular steps and which are going out from this particular step okay for that we are going to gather this particular, restricted graph.

Now see here and all the states are leveled because these are the steps will be calculated for easy pay there exist a part no where P holds if any step is not true then this true position of having EGP at the particular step. So you are restricting the graph. Where only P is true, now next what we are going to set up find a strong,

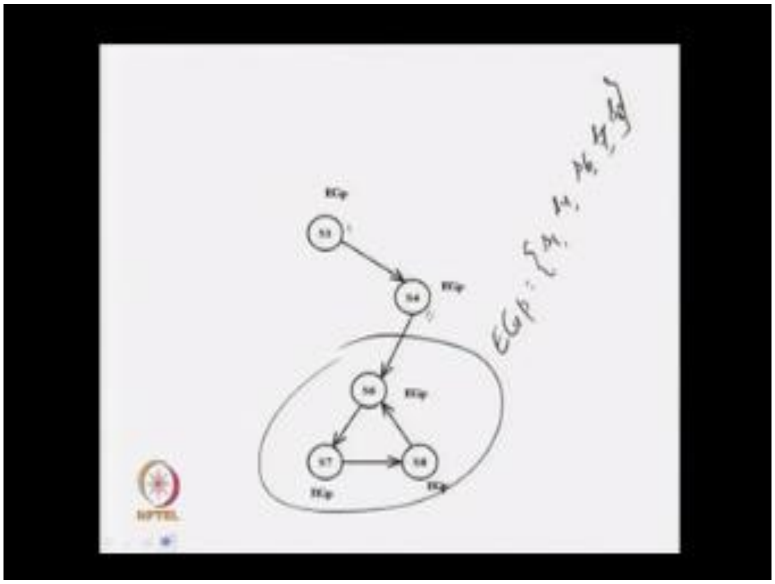
(Refer Slide Time: 50:01)



Connected component now what is a strongly connected component it is the maximum reason where we can get each step from any other step, now if you look into this particular step where you are going to get this particular portion as the maximum strongly connected component. Now it is having S6, S7, and S8. You see if I am in S6 then I can go to S7 I can go to S8 if I am in S7 I can go to S8 and go to S6 from this particular point.

And if I am in your S8 then I can go to S6 and S7 so it is satisfied whether I am in any state in can go to any state but now if you are in S4 then you cannot go to S2 okay. So that is why S4 is not coming secondly if it is considered S4 from any one these step I cannot go to S4 so it will not come from so I am getting these as a strongly connected component.

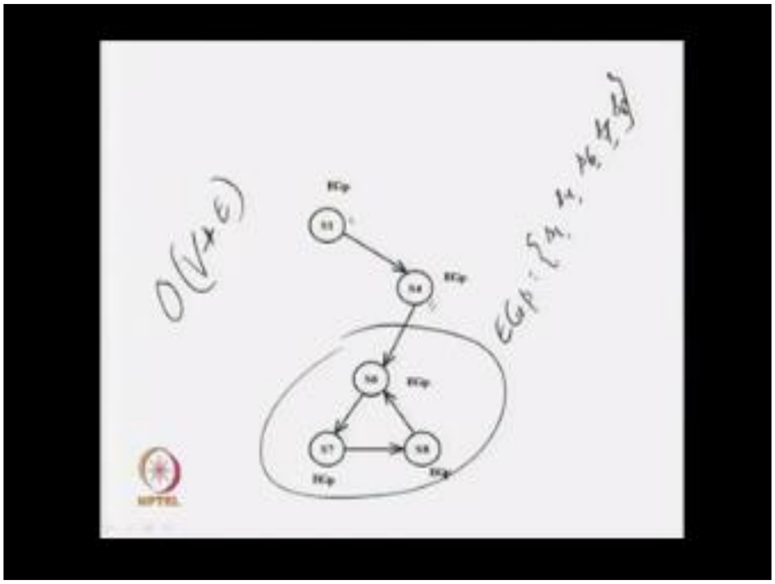
(Refer Slide Time: 51:28)



Now what we are going to do now we are going to have now a equal break point from this particular strongly connected component and if any step can be by it is point. Now from your EHP will be true. When you are starting from this place for break for so we can raise your P and move for then I can reach that one. But from here I cannot go then I cannot reach this particular S2 because this is a directed. So okay in this way these two steps will come in to place that means you itself see that from your this maximum strongly connected component.

We can use break on to reach for and this S1 and no more steps will come so after that whatever we can reach from the particular maximum strongly connected component those will be having EGP2 now wherever you go you will find that P is true because this is eventually we are coming to this particularly strongly connected component and wherever you go you will find that P is true. So these are the steps where EGP is true. So similarly we are getting that EGP will be true S1, S4, S7, S8.

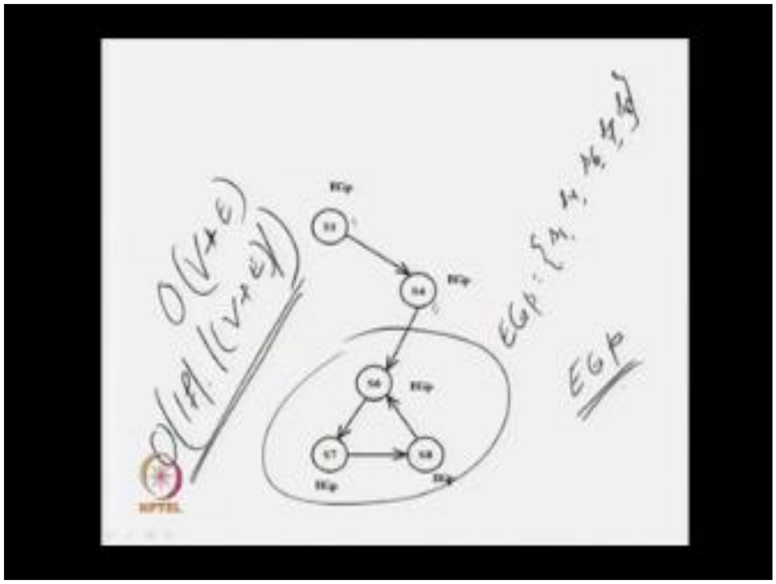
(Refer Slide Time: 52:49)



Now in this particular case we just see that what the content we are getting over here and what we are getting we just see that what will happen since we are getting particularly strongly connected component and along with that the break of are we are having algorithm which can be done in any other which is proportional to my step number of particles and number of ages okay. so we are having algorithm for your break of and finally strongly connected component which works in all your model of your P okay.

So that is why we are having algorithm works in order in P process what happens in this particular case the models algorithm we are getting are time complexity depends on the length of the formula and your size of your starts steps or size of your graph okay. This is some sort of algorithm so eventually complex A will be like that so EGP now what complexity you are getting it is linear on the length of the formula and it is linear on the size of my graph in case of your EU and your AF.

(Refer Slide Time: 54:56)



The complexity is basically liner in the length of the formula but nit is quadric in the size of the step that is aquatic in the number of notes we have in my state space okay. so that means you see that instead of A because AU I cannot about EU I cannot abort AU I cannot about because sorry I can say EU but I cannot avoid I may have an operator AU it has to expressed with the help of EU. Okay now I have this AF but okay we are having instead of AF I can go for EG okay because you know that I am having.

So you can look for one complementary set of operator EX EU and AF this is one o0f the operators you know the operator I can say EX EU and say EG. Okay now you can look for this is similar so if we are going to have these things then so that I can say it is Ex so if I am going to look for these three operators then what will happen that time complexity will be slightly less. And if you are going to look for these three sets then the time complexity will be slightly more because it is quadric in the size of my document.

So this one now I can go for this particular procedure or go for these particular step also basically we are going to look for three set of operators and others will help of these two operators so this is the another operators we have seen and we have seen that time complicity may vary place to place okay. With this I will stop here today in next class we will look for some examples where we are going to have a make model you are going to see the property is satisfied model 3 algorithm 2 does it.

Acknowledgement

Ministry of Human Resource & Development

Prof. Satyaki Roy

Co-ordinator, NPTEL IIT Kanpur

NPTEL Team

Sanjay Pal

Ashish Singh

Badal Pradhan

Tapobrata Das

Ram Chandra

Dilip Tripathi

Manoj Shrivastava

Padam Shukla

Sanjay Mishra

Shubham Rawat

Shikha Gupta

K. K. Mishra

Aradhana Singh

Sweta

Ashutosh Gairola

Dilip Katiyar

Sharwan

Hari Ram

Bhadra Rao

Puneet Kumar Bajpai

Lalty Dutta

Ajay Kanaujia

Shivendra Kumar Tiwari

an IIT Kanpur Production