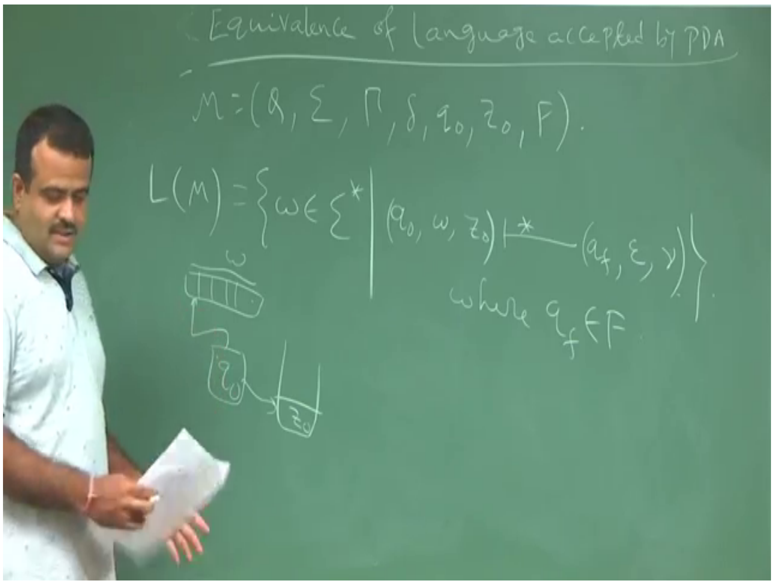
Introduction to Automata, Languages and Computation

Prof. Sourav Mukhopadhyay
Department of Mathematics
Indian Institute of Technology, Kharagpur

Lecture – 53
Equivalence PDA

So, we are talking about the language accepted by; the equivalence between the language accepted by final state and the language accepted by empty string.

(Refer Slide Time: 00:26)

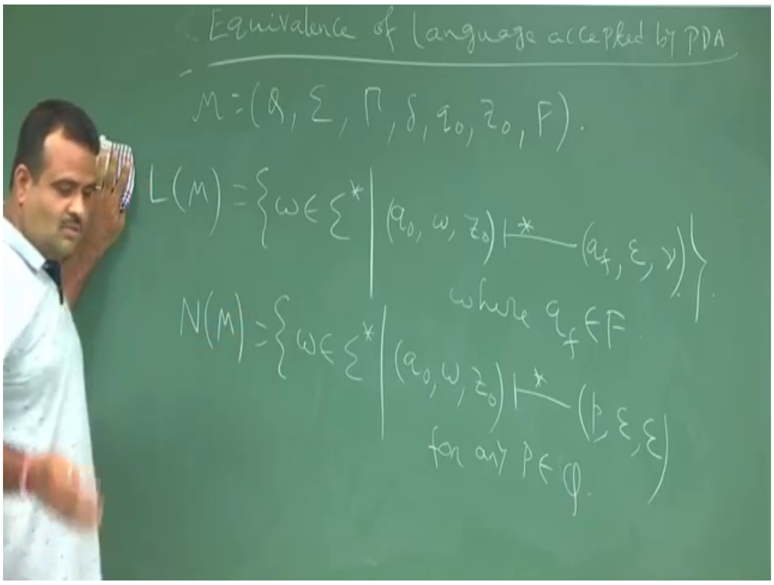


So, just to recap given a PDA, we have this tuple Q, sigma, delta, q 0, z 0, F. Now, this is the set of all states and this is the finite set of input alphabets and this is the symbol which we are going to use in the stack, stack symbol basically and this is the transition rules. And this is the starting state q 0 and this is the starting stack symbol and this is F, the set of all final state.

Then we know the language accepted by this PDA, in the means of final state accepting by final state is set of all strings such that we start with q 0 with this w in the tape. So, q 0 is the starting state and the tape contained w dot this is our w. So, it is pointing here and the stack contained z 0. So, stack is z 0 if this move with the different I mean, with the iterative applications of delta; if it is going to a final state at the end of this execution at the end of this tape. So, that means, if it is going to some q of f and this is epsilon and we do not care about the stack symbol because this is the language accepted by the means of final state where q f is a final state ok.

Now, this is the language accepted by final state and now we know another way of acceptance that is the language accepted with the means of empty stack.

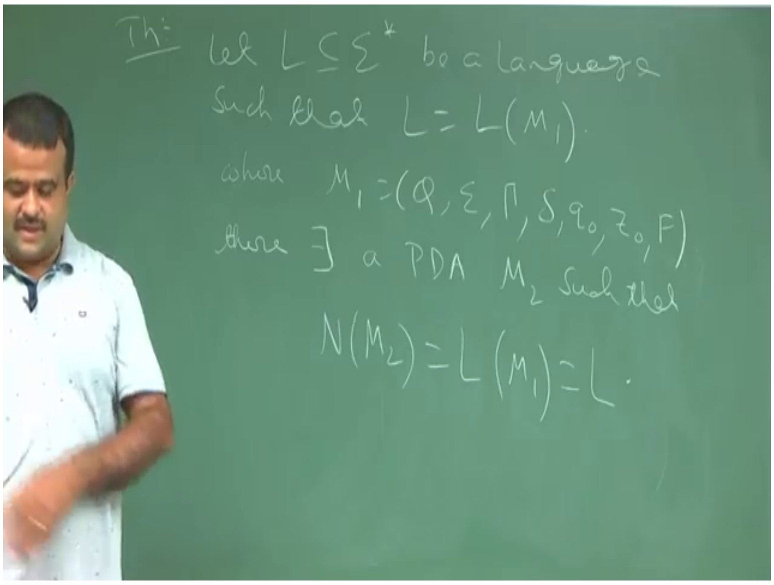
(Refer Slide Time: 02:36)



So, that is nothing, but so, this is set of all w such that we start with the same positions q 0, w, z 0 and we end up with a p. We do not care about final state over here because it is just a in the means of empty stack. So, stacks should be erased; for any P belongs to Q. So, here we do not need the; we do not need to reach to a final state. It can be any state, only thing is required is the stack should be empty. So, all the symbols would be erased from the stack ok.

Now, we will see the equivalency between these two. In the last class, we have seen if the language accepted by this, then we have a corresponding PDA which can accept by the means of L of means of accepting the final state. So, now, we will today we will discuss the; if we have a PDA which is accepting a language by the means of final state, then it has a corresponding it will correspond, then we can construct a PDA which will accept the same language by the means of empty stack. So, let us try to do that ok.

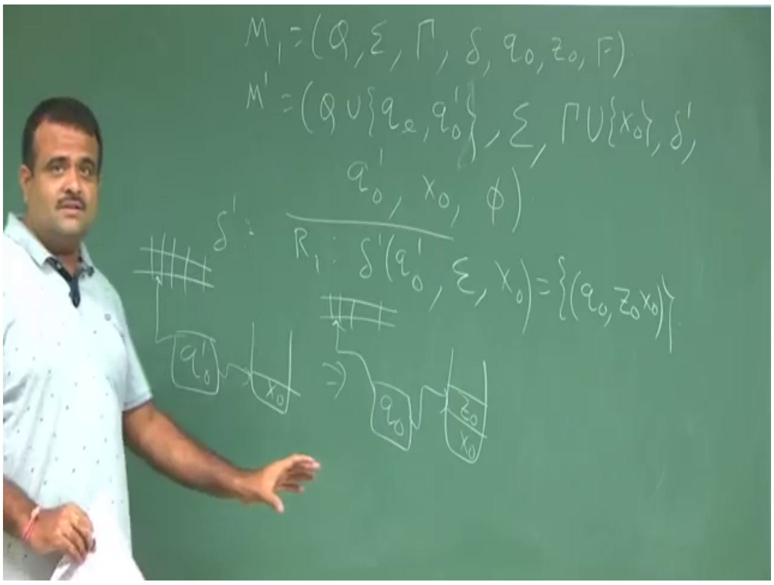
(Refer Slide Time: 04:30)



So, you want to write this in a theorem, let L be a language such that L of M 1; that means, M 1 is a PDA which is accepting this language by the means of final state. So, that means, say M 1 is a where M 1 is a PDA say Q 1 or say Q, q 0, z 0, F give it this PDA.

Now, now the theorem is telling, then we can construct a PDA M 2, then there exists a or we can construct a PDA pushdown automata M 2 such that the M 2 will accept the same language, but by the means of empty stack. So, this we have to construct. So, M 2 will be accepting the same language by the means of empty stack ok. So, how to construct M 2? So, proof is as follows; proof is as follows.

(Refer Slide Time: 06:24)



So, we have given M 1 ok. Now, M 2 we will take help up two extra states. So, this is our M 1. In the lecture note, this is given by M 2 anyway I mean notice on does not make any difference. So, we can take them M prime if you like. So, this is basically Q of will take help of two extra states; this is the new starting state for this automata and this is the state with which we are going to use for reaching to the final state. I mean once we reach to the final state will keep on hop there and either keep on erasing the stack. So, we will do that.

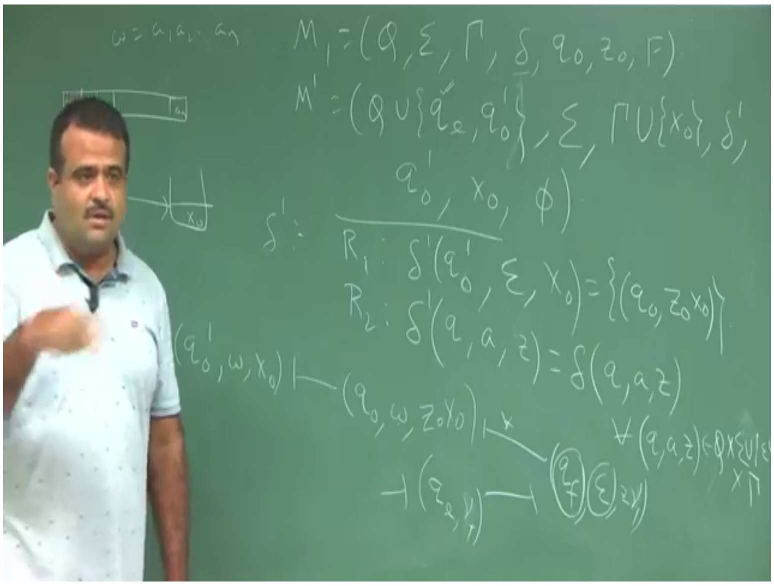
This sigma is same and input stack symbol we are taking help of our new stack symbol X 0 and which we are going to use for the starting state starting stack symbol for this new automata and delta prime delta prime we have to define and q 0 prime is the starting state for this new automata and X 0 and this is the empty stack. So, we do not care really care about the final states so because this is we are accepting this by the means of empty stack ok.

So, how we can do that? So, let us just fire define the delta. So, this is the M. Now, we have to define the delta hat. So, delta hat what we want to do? We want to first initialize this put the X 0 there and this z 0 there. So, delta hat is like this. So, we have a rules like this R 1 so, delta hat of q 0 prime. So, we are going to start with q 0 prime with epsilon, this is the determine a non-deterministic PDA and the stack symbol is X 0. Now, these will initiate the process of this M 1. So, it should initiate the process of M 1.

So, this should contain this q 0, then we have z 0, X 0 ok. So, without reading anything we are starting with q 0 prime and our X 0 is here. So, what we do without reading anything in the tape symbol, we just move our process to this one that q 0 which is the starting state for this given automata and in the stack symbol we will add this z z 0. Now z z 0 is the top of the stack which is the now this is the pointing symbol of this system and then we have a now, we are because that is with epsilon moves. So, we are not reading anything. Now, we are going to start reading this. Now, we follow the rules of this M 1 ok.

Now, we reach to M 1. This is the initialization process to reach to the system M 1. Once we reach to the M 1, then we will start applying the rules of M 1. So, that will write. So, this is just to bring it to the M 1.

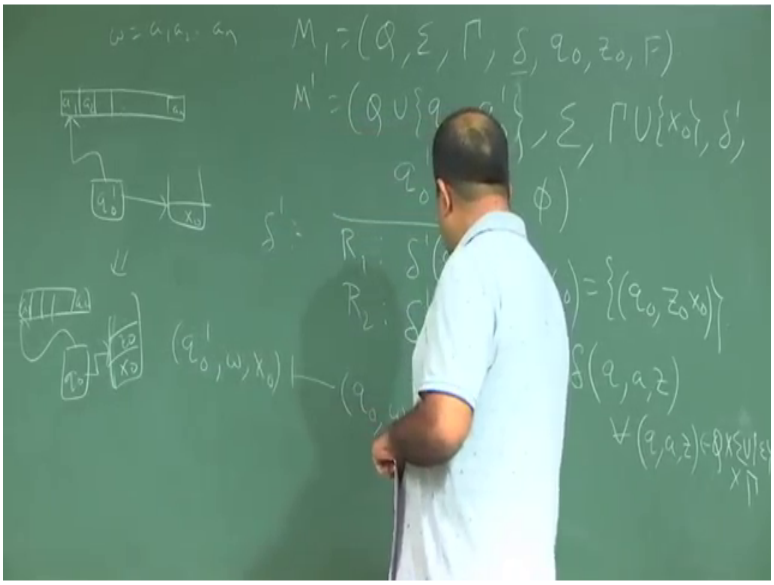
(Refer Slide Time: 10:42)



So, now we have delta prime of any q, a and any z which is same as, now we are at M 1 it will be delta; this delta, delta of q, a, z and the same z and this is true for all such q, a, z which are coming from q is coming from Q and a is coming from sigma star; we can say because we have epsilon move also not sigma star sigma star is the string. So, sigma, including epsilon and the stack symbol ok; so, this is just a rules of; this is just the transition of the given PDA ok. So, we just convert this initial state to this given PDA.

Now, we have rules for PDA. So, that will continue. Now, once they w will reach here. So, we will have this. So, first we are starting with q 0, w and z 0 sorry X 0. So, now, it will leads to what? It will leads to q 0 with epsilon move we are not using w. So, this is the situation w is here say w is say a 1, a 2, a n. So, this is the tape a 1, a 2, a n. So, we are starting with q 0 and our stack symbol is X 0 this is pointing here and it is pointing here ok.

(Refer Slide Time: 12:52)



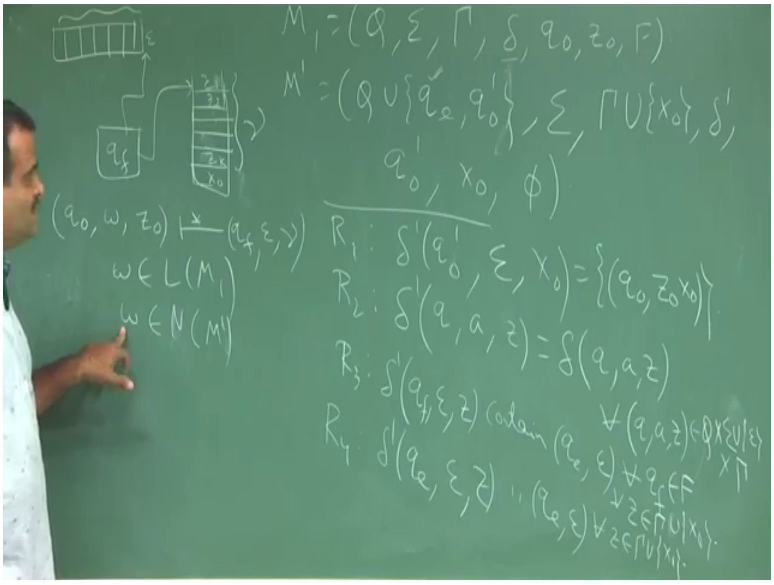
Now, without reading anything we reach to q 0. q 0 is the starting symbol of this starting state of this original PDA and then this will be z 0, X 0 because z 0 is the starting symbol in the stack of the original PDA. So, it will reach to. So, this will reach to here, this will reach to we are at q 0 now and this tape is remain same a 1, a 2, a n and but the stack is we have X 0, z 0. Now, we are pointing z 0 and it is pointing here.

Now, we have a z 0 on the top of the stack. Now, we have a rules for now once we reach here, then we will apply the rules for this and if we keep on apply the rules for these. If the out w is accepted by this if w is accepted by this PDA in the means of final state so, what it will where it will go? It will go to some final state of M 1 and this will be finished and we do not care about this.

So, now these we have to go for another, we need to write the delta hat so that it should reach to the empty stack sense so, that that we are going to write. So, then once we see that we are reaching to a final state and this is m epsilon, then we move to a state, then this state we are going to use it that we have to write. Then we will move to a state like this q e, q e, epsilon and then again from q e; no q e, epsilon. So, epsilon means if this is say some z gamma 1. So, this will be epsilon means gamma 1 so like this.

So, let me write this rule 3 then it will be more clear. So, idea is once we reach to any of the final state of this given PDA, then from there we will go to the some intermediate state which is q e and from there we will keep on erase the stack symbol so that it will reach to a final state I mean it will empty the stack symbol by repeatedly. So, let me write that, I just explained the idea. So, let me complete the rule.

(Refer Slide Time: 15:35)



So, R 3 is basically R 3. So, delta hat of q, epsilon, z for any Z, this contain q e, epsilon for all we can write f q f for all q f belongs to F and for all z belongs to this in your along with the X 0 ok, this along with X 0. So, this will keep on doing.

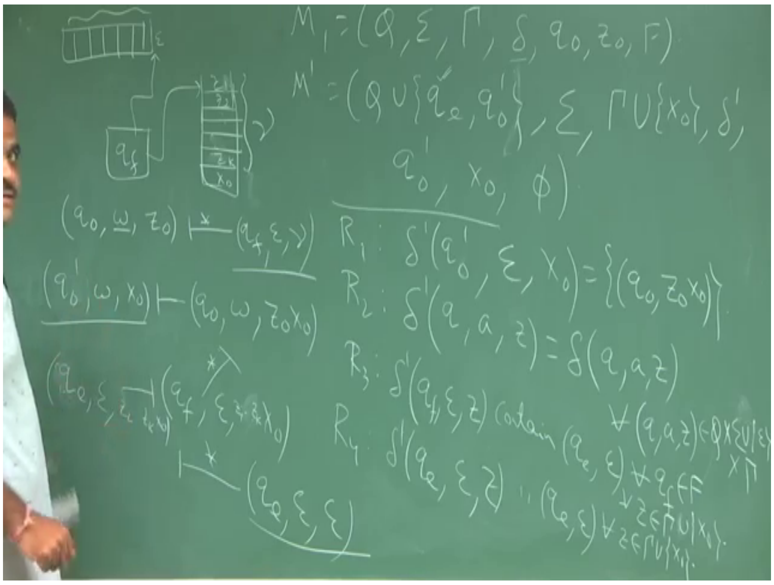
So, now we are reaching there. So, from there we will keep on do this. So, that we have to write delta hat of once, we here in q e; q e is the state which are going to use for erasing the stack symbols, that is all. So, q e, epsilon, z this will be remain at q e contain q e, epsilon; this is true for all z belongs to this ok. So, that is the idea, these contain this.

So, that means, once we reach to q e; let me just explain. So, suppose our this is empty epsilon I mean nothing is there it is done and we reach to the some of the final state and we have still we have some symbol in the stack, say it will be X 0 will be remain here z

1, z 2 or we can say like yeah. So, dot dot z k. So, sorry z 1, z 2, z k and suppose at this point of time it is pointing here and this is our gamma.

So, that means, what? That means, this is our w up to these are w. So, this w is accepted by this PDA in the terms of final state. So, that means, q 0 of w and this z 0 will reach to will reach to some q f, this is epsilon and this will be gamma. So, sorry this is gamma ok, up to this. Now, we want to this is. So, this is belongs to L of M 1. Now, we want this w should belongs to L of M prime sorry N of M prime. So, that means, this w should be a string which is the accepted which should be accepted by this new PDA by the means of empty stack. So, how this will this transaction will help us?

(Refer Slide Time: 19:14)

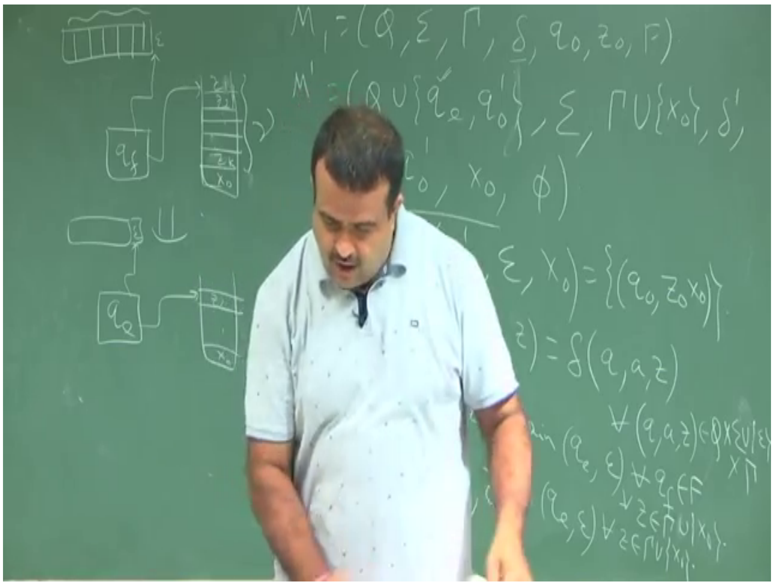


So, this we are starting for the new, we are starting with this; we put w we put X 0. Now, in the first step we are going here. So, q 0 then this will be remain same because we are just reading epsilon. This is epsilon, then the stacks is this ok. Now, we will use this rules because w is the accepted state, sorry accepted string accepted in the means of final state. So, it will by the move relation it will move to this situation this ID, this instance in your description.

So, what do we do? We just keep on apply this and then it will reach to like this we will use this q f, epsilon and this z 0 this will be gamma X 0. Now, gamma is z 1, z 2, z n. So, this is nothing but so, now, we will apply this rule R 3. Now, once we apply R 3 this will go to q e. This will go to q e epsilon and then this is gamma is nothing, but z 1, z 2, z k.

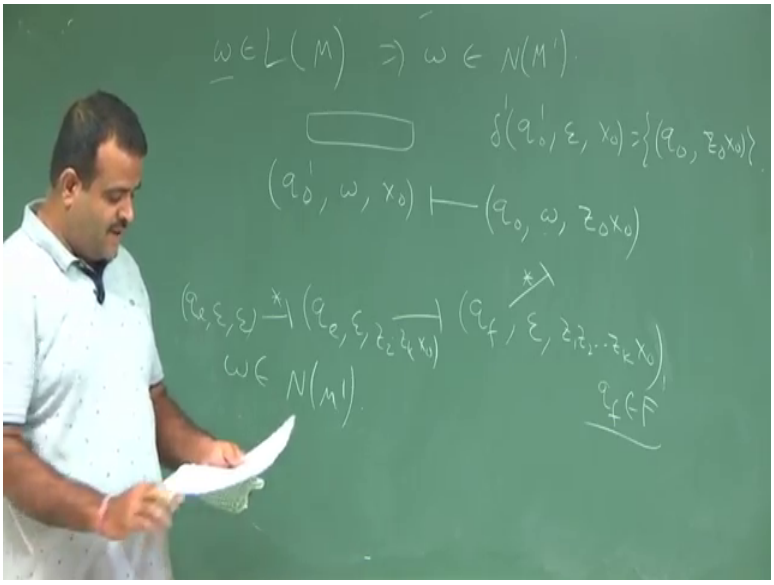
So, z 1 will vanish so z 2, z k, X 0 like this. So, this could be epsilon also this gamma, then we have X 0. So, for X 0 also we have this rule and this will keep on doing. So, this will keep on doing by applying this. So, ultimately this will be going to the situation like this q e and epsilon and it will be epsilon. So, this initial idea is going to this final idea and from here you can say this is accepting this; that means, we are just reaching to the situation like this.

(Refer Slide Time: 21:29)



This is after q f we are shifting it to q e and this is this is ok, this is gone already gone. This is epsilon is nothing is there actually epsilon is nothing empty and now this z 1 we have erased. So now, z 2 like this X 0. So, we will keep on do this using this rule 4 until it got empty. So, this erasing process so, this q e we are using for erasing this stack symbol ok. So, this is the idea. So, now we will see the yeah we can this is the proof like let us write the formal way. So, I will just give you the now idea of the proof. So, this is the construction.

(Refer Slide Time: 22:50)



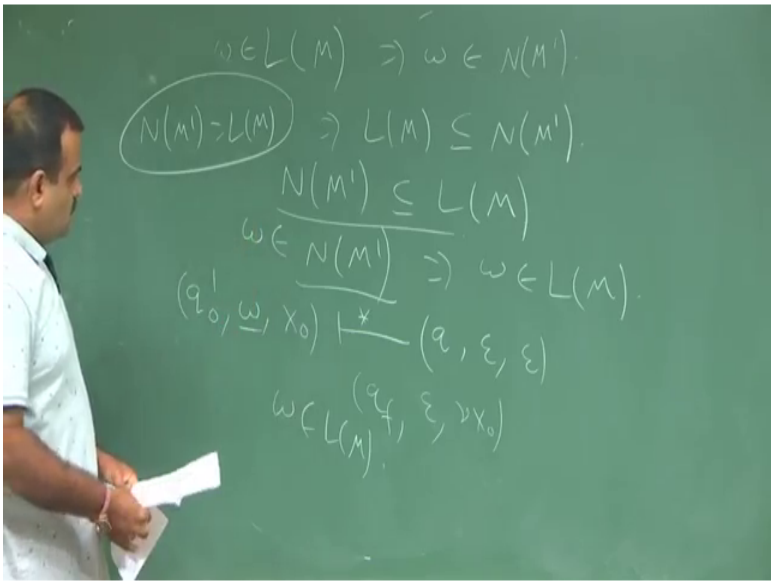
Now, yeah so, we can just so, we have just now we have seen that L of M if a w is belongs to L of M this imply w will be belongs to L of M prime. That means, so, this just now we have seen, we have taken a w which is q 0 prime, w, X 0. Now, if this w we can go to this is the first step with the epsilon move z 0, X 0. This is we that we are using this rule that q prime of q 0 prime, epsilon and X 0 this will contain q and this will be z 0 X 0.

Now, since w is the acceptance state acceptance string accepting by the final state so it will reach to by repeated application of delta of the original PDA it will reach to some of the final state and some z 1, z 2, z k, X 0 and this could be empty also. This could be epsilon also, we do not know. Only thing we know q f is the final state of this.

Now, we apply the rules of this delta hat. Now it will keep on, it will fast it will go to this state and it will keep on release this. So, in the next step it will release it will put epsilon like this or finally, it will go to q e, epsilon, epsilon. So, this is the proof; that means, w belongs to N of M prime ok.

Now, the other way around; suppose, we have a w which is. So, this implies that L of M is a subset of N of M prime M prime.

(Refer Slide Time: 25:05)



But, we want to show this these two are same. So, further we need to show L of M prime is a subset of L of M ok. So, that means, if a so; that means, we can take a w from here L of M prime then we have to show this w must belongs to L of M ok. So, how to show this? So, this is also quite trivial to show this. We can just ok. So, how to show this?

So, like w is accepting this in the so that means, we are starting from q 0, and then w, X 0. So, this w is accepted by this means, it will reach to state like this some of the states we do not care about that. So, some q and yeah some q, but we are following the rules of this. So, some q and then epsilon, and this will be epsilon.

So, then we can use the yeah; we can use the R 2 let me check it out. So, once we reach to epsilon yeah so, how to justify this w is yeah. So, we are reaching to epsilon means in the middle must be we reach to this state. The q f and then this is epsilon and this has some gamma X 0; gamma could be epsilon also. And, from there we can apply our rule of this delta and we can reach to a final state of this. So, we already reached a final state. So, that imply w belongs to L of M ok.

So, that; that means, L of M is a subset of this. So, this means L of M prime is equal to L of M. So, this is the proof. Anyway, the proof formal proof will be given in the lecture note.

Thank you.