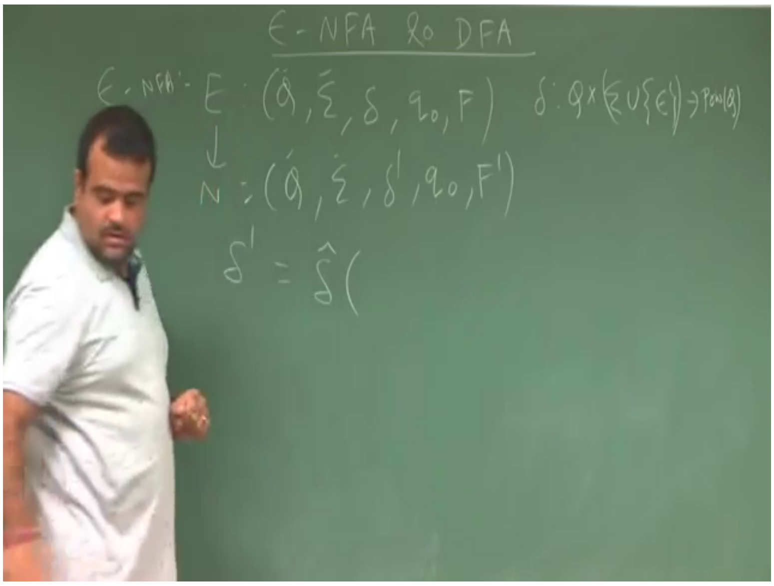
Introduction to Automata, Languages and Computation
Prof. Sourav Mukhopadhyay
Department of Mathematics
Indian Institute of Technology, Kharagpur

Lecture - 15
Epsilon-NFA to DFA

So, we are talking about epsilon NFA conversion from a Epsilon NFA to NFA then the DFA. So, now we will discuss how directly we can construct a DFA from a epsilon NFA. So, this is a little variant of the subset construction.

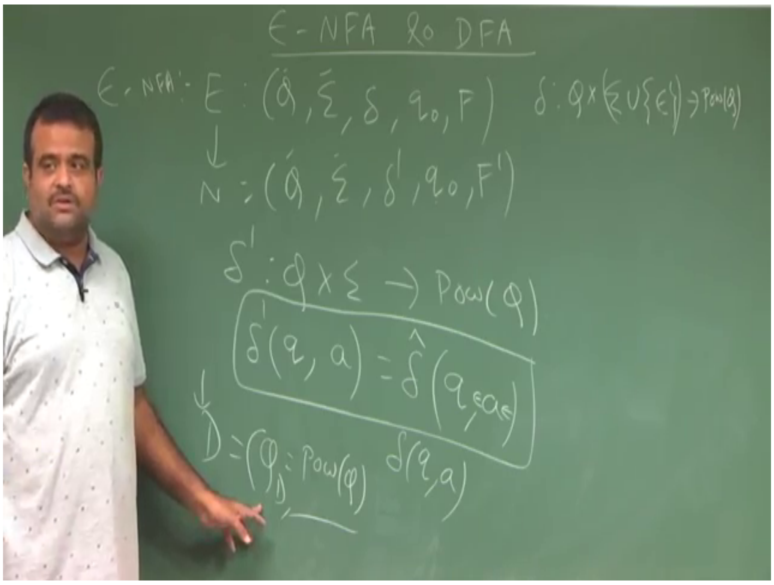
(Refer Slide Time: 00:37)



So, just to recap so, we have given a NFA epsilon NFA E, delta, q 0 f and from here we can have a NFA this with the same set Q, delta prime, q 0 is same F prime ok. So, here delta is the function form Q cross we are taking the epsilon move to power of Q ok.

So, now this is the set is same Q, this is same set of alphabet is same, now here is a change, here is the change here this is also same. So, delta prime we are defining this is the transition rule for this NFA, we are defining as this delta hat of delta prime. So, delta prime is a function from Q cross sigma to power set of Q.

(Refer Slide Time: 01:49)

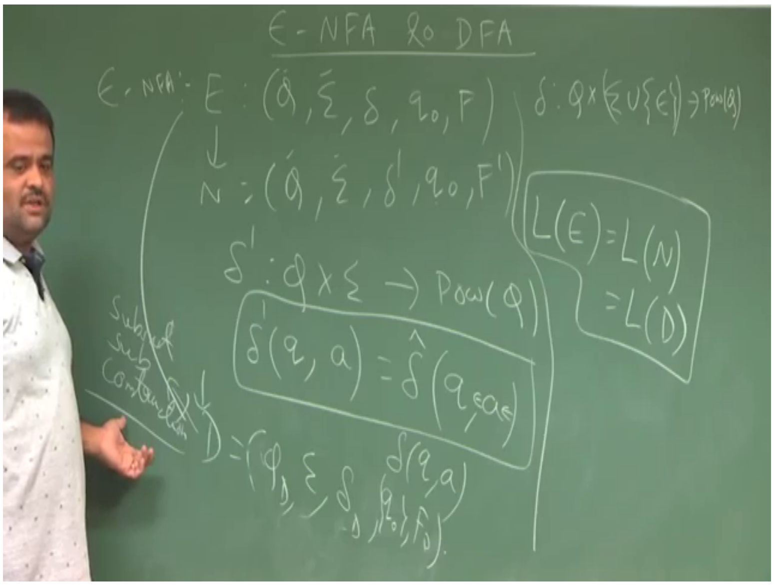


So, it is not allowing the epsilon move here for this. So, we have to take care of epsilon move before a or the after before a given symbol. So, that will be taken care by this. So, delta prime of a state q and a alphabet a will be delta hat, this delta hat of q comma a this is the way we define the transition rule for the NFA. So, this means this when you talk about hat there is a difference between in this delta there is a difference between hat and this. So, this is this will give us a set of states, but it is not involve the epsilon move.

But here hat means we consider a as before a there is epsilon after a there is a epsilon because epsilon is just empty string. So, you have to consider that move also. So, that will be taken care by the extended transition function which is hat of this. So, this is the way we defined this and then from here so this is our NFA from the epsilon NFA we have constructed the NFA and from the NFA we know we can construct a DFA which is Q D which is basically the subsets of this. But not we no need to consider all the subset because some of the subsets are not reachable from the starting state. So, those we will not consider, those are called date state ok.

But in the worst case if there are n state it may happen that 2 to the power n is the size of this. But most likely in the many cases I mean mostly there are only same number of states will be here, I mean we do not need to consider all the 2 to the power n states ok. But anyway theoretically it is the power of Q, but again we do not need to consider all the states, all the subsets because few of the subsets are not reachable for the starting state.

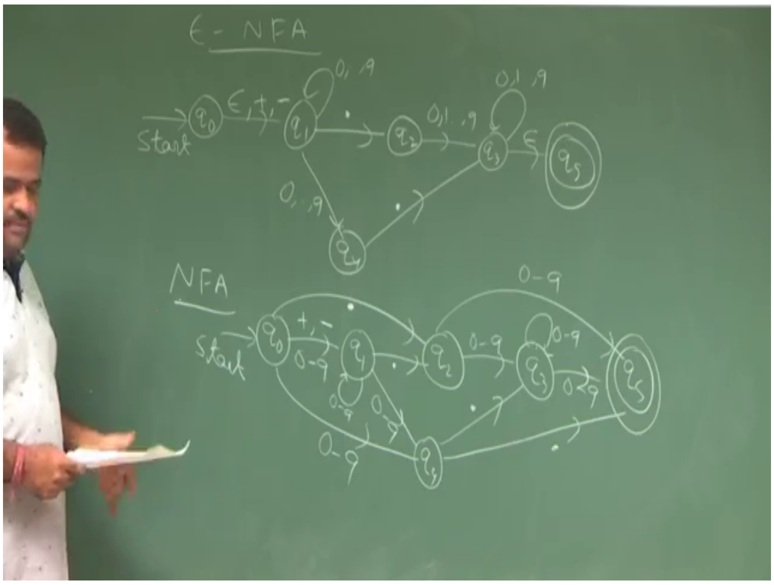
(Refer Slide Time: 04:06)



So, this Q D sigma is same delta D transition rule and this q 0 this subset and F, F is F D. So, this is the construction called we know this subset sub construction. Now, we can prove that if a language is accepted by this so, this is d. So, language accepted by this epsilon NFA is same as the language accepted by the NFA which is same as language accepted by this DFA. So, they are basically equivalent; so they are equivalent. So, any language I mean a language is called regular language if there is a finite automata which accept that, accept all the strings of that language. So, if we have a given a language we need to state test whether this is a regular or not, we can because this NFA is easy to construct epsilon NFA.

So, we can just try to construct a epsilon NFA for that language and if we can show that that language is same as the language of that epsilon NFA then we are done because we know that there is a corresponding NFA and there is a corresponding DFA which accepting that. So, this is the way we can do that. So, this is the so now, we will see how the directly we can come from this epsilon NFA to DFA in set of via this NFA that is also a subset construction a variant of substrate construction, so that we will see now ok. So, for that let us take the example. So, let us take the example first, how we.

(Refer Slide Time: 06:27)



So, suppose we have a given epsilon NFA. So, this example we have seen q 0 this is the starting state and we have q 1 which is epsilon and this plus and minus and from q 1 we can see 1 to 9 that is digit it will be remain in this.

Now, if you see a dot will go to q 2 and from q 1 there is another move to q 4 if we see 0, 1 to 9 digit and here is a q 3 which is again move by the digit from q 2 and if you see a dot from here we can we will go to q 4 to q 3. And then we at q 3 we will have a self loop with this input on the tape as digit and then we will reach to the final state q 5. So, this is our this is again by epsilon move ok.

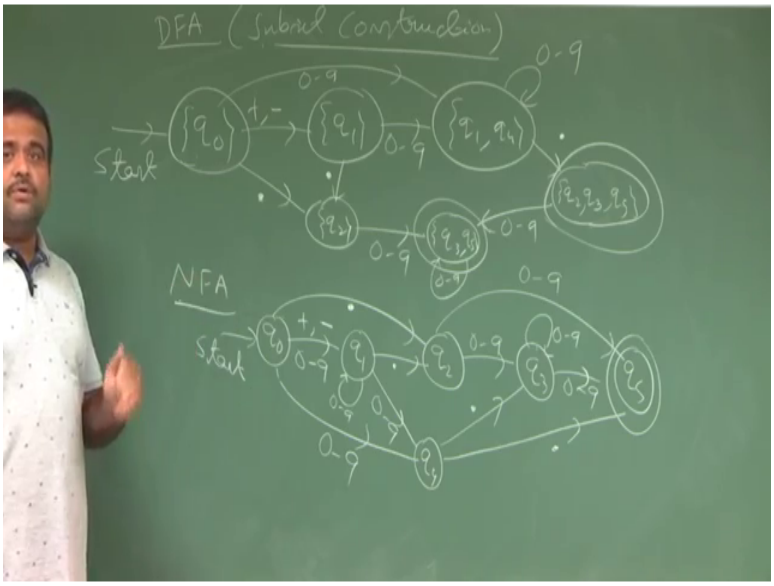
So, this example we take we have we have seen this example before also. Now from here this is epsilon NFA; we in the last class we have constructed the NFA and the DFA corresponding DFA. So, the NFA we constructed is like this, this is just to show you the difference if we go to directly. So, the NFA was like this. So, we for the NFA we just need to plug the epsilon move nothing else. So, states are all same so this is q 0, q 1, q 2 and this is q 3, q 5 is here and we have a q 4 over here.

So, now this is the starting state and from here we can go to plus or minus we can go to q 1 or 0 to n also we can go to q 1, with dot we can go to this example we have seen in the last class we have just copying. So, this is from here we can go from 0 to 9 any digit we go here now with any digit we can go 0 to 9 or with any digit we can go from here to here and with dot we will go to.

So, we are just plugging the, absorbing the epsilon move with the normal string. So, this is the way and then we move to the from q 2 we go to the q 3 with 0 to 9 or directly we can go to the final state q 5 with the digit 0 to 9. And from q 4 with dot we can go here or we can go there and here is a self loop 0 to 9 and here also 0 to 9, any digit between 0 to 9 ok.

So, this is the corresponding NFA if this is epsilon NFA this is the corresponding NFA. Now we have DFA also, in the last class we have seen that DFA. So, given a N epsilon NFA we have constructed this NFA, from this NFA we will construct a DFA.

(Refer Slide Time: 10:24)



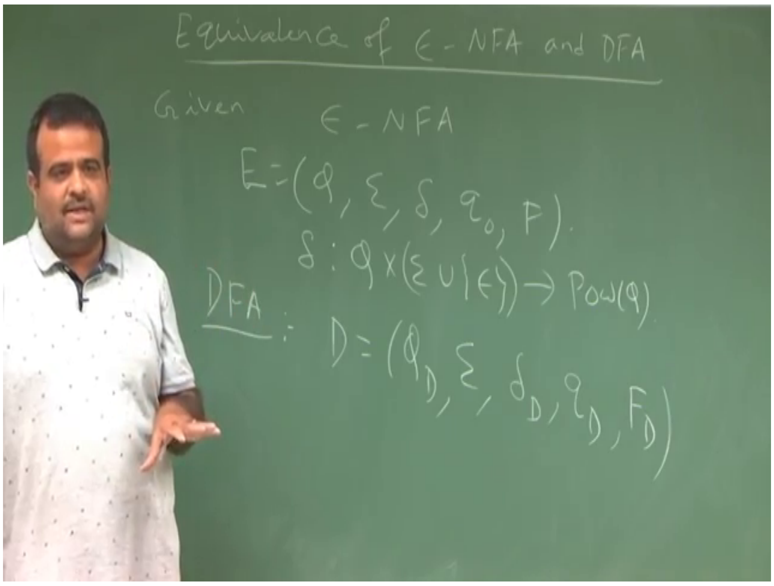
This is same as that a subset construction ok; so, these also q 0, so we considered so from here to here. So, we have to consider all the possible subset of this and we consider those subset which are reachable. So, we had that delta this transition rule table. So, I will just draw the diagram here. So, transition rule table is there in the previous lecture. So, q 0 q 1 these are the possible state we can go from the starting state and here q 1 q 4 and here q 1, q 3, q 4, q 3, q 5 and here we have q 2, q 3, q 5 q 2 q 3 q 5. So, these are the possible state we can go from this.

Now, with this is the starting state now we have to write the transition rule, now with. So, from here to here we can go with plus minus dot any one of this oh no sorry dot not plus and minus. With dot we are going to q 3 and from here to here, we go from here to here if we see any digits 0 to 9 and here if we see 0 to 9 will go this. And if we see a dot we will come here and from here we again if we see 0 to 9 we will go there and here also 0 to 9 we hop there. And we dot we come here and with 0 to 9 you go here and 0 to 9 you hop there ok.

And this is these two are the final state because final state content all the final state of this. So, q 5 is the final state so whatever in the subset q 5 is there those will be the final state. So, this is the corresponding DFA from this NFA. So, we first constructed the NFA from the epsilon NFA then we construct the DFA by the subset sub construction. So, this transition rule we have written in the last class.

So, now we will talk about directly from the epsilon NFA how we can go to DFA. So, that is a variant of the subset sub again. So, that part we will do now. So, given epsilon NFA how we can reach to the DFA directly, equivalence. So, this is called equivalence of epsilon NFA and DFA.

(Refer Slide Time: 13:46)

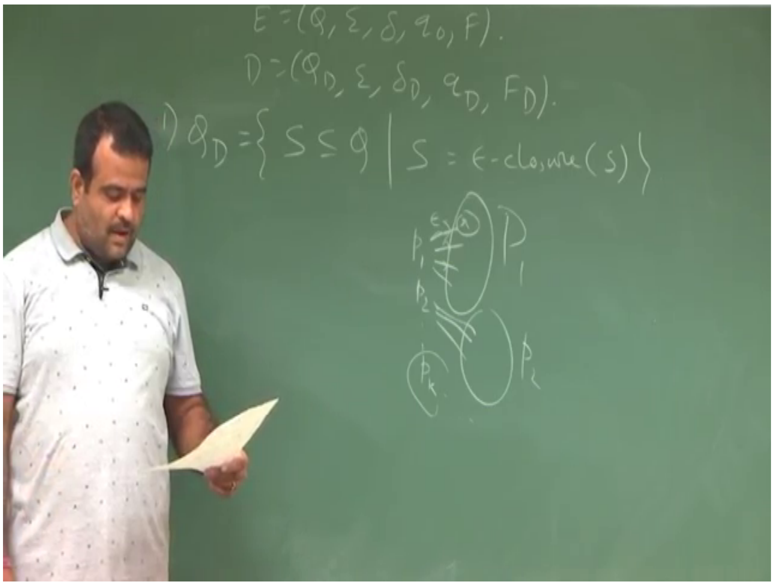


Epsilon NFA and DFA ok; so, it is a theorem we can consider they given a language, a language is accepted by the same epsilon NFA if and only if the language is accepted by DFA. So, DFA can consider as a epsilon NFA because epsilon NFA is not necessarily in that we will have a move for epsilon, there is optional that is optional. So, DFA can be considered as a epsilon NFA.

Now, given a epsilon NFA how we can construct a DFA directly that we look at. So, given a epsilon NFA given epsilon NFA say E, Q, sigma, delta, q 0, F where delta is the function from q cross sigma union epsilon move to 2 to the power q. So, given this we have to construct a DFA. So, for DFA we write this as Q D sigma will be same because our tape is input tape is same. We put everything in the tape and then we keep on reading that a 1 by 1.

So, this is same rule will be changing because this Q D is changing and obviously, starting state will be changing and the finals acceptance state will be changing. So, we have to define all these. So, this is same there is no change in the input alphabet, this is a finite set. Now Q D, what is QD? So, normally in the DFA from the NFA what we take we take the all the possible subsets, but here we are taking their closures also ok. So, we will take all the epsilon closure of the subsets, that is our state. So, how we define that? So, let us just erase this I will write it again.

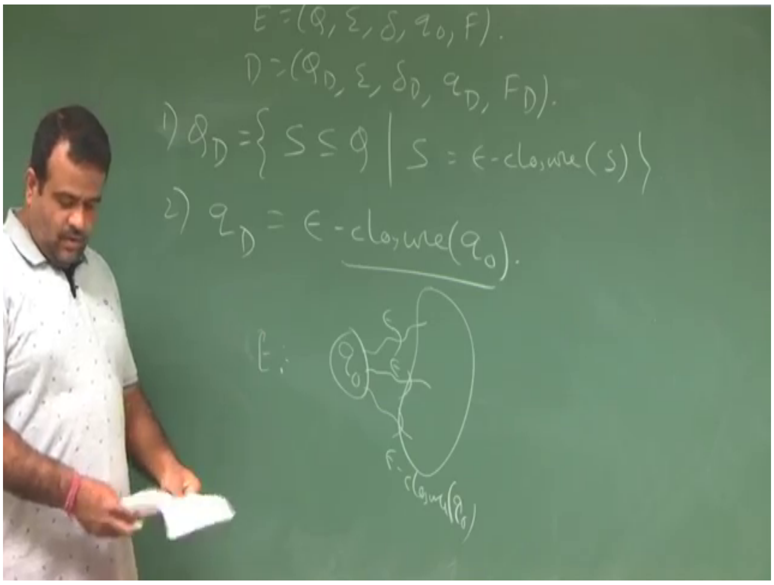
(Refer Slide Time: 16:31)



So, we have this NFA ok, now from this we are constructing the DFA ok. So, first of all what is Q D? Q D is the subset of Q or you can say Q E if you like. So, subset of Q such that S is epsilon closure of S. So, we just considered we take a subset we take a subset say p 1, p 2, p p k we take a subset. Now we consider epsilon closure of this, epsilon closure means by the epsilon move we can go to this. So, this is our capital p 1 say. So, similarly epsilon closure of this is our capital p 2 say epsilon closure of this is our capital p n.

So, we take those union of this as our Q D. So, Q D is now the epsilon closure of the subsets earlier form NFA to DFA it was only the subsets, but here we are taking the epsilon closure of those states ok. So, we have a subset, subset is consists of a states. So, we consider epsilon closure of those states. So, that forms the Q D ok. Now, this is the Q D I so, this is the Q D.

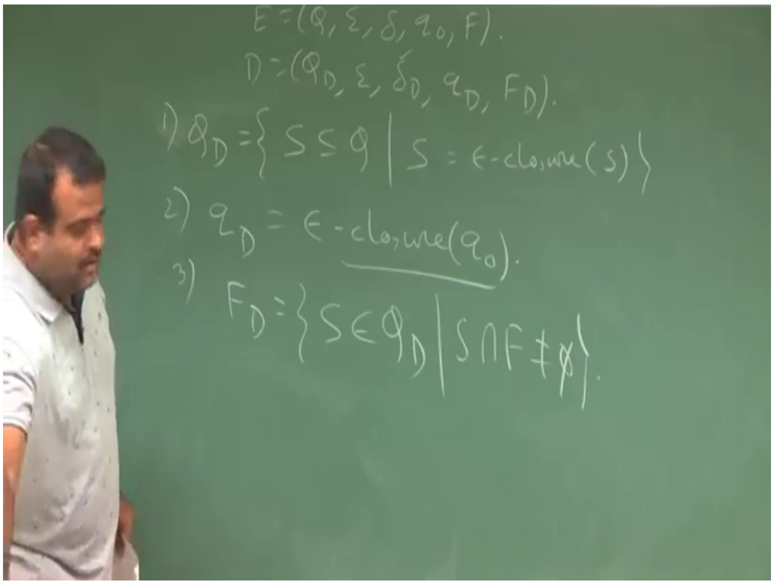
(Refer Slide Time: 18:28)



Now we consider the starting state starting state is nothing, but epsilon closure of q 0 this is our q 0 in the epsilon NFA in the epsilon NFA this is our q 0 from this q 0, but the epsilon closure means by the arc with only epsilon level. That means, epsilon move if we can reach to so this set these are all this is the set epsilon closure of q 0.

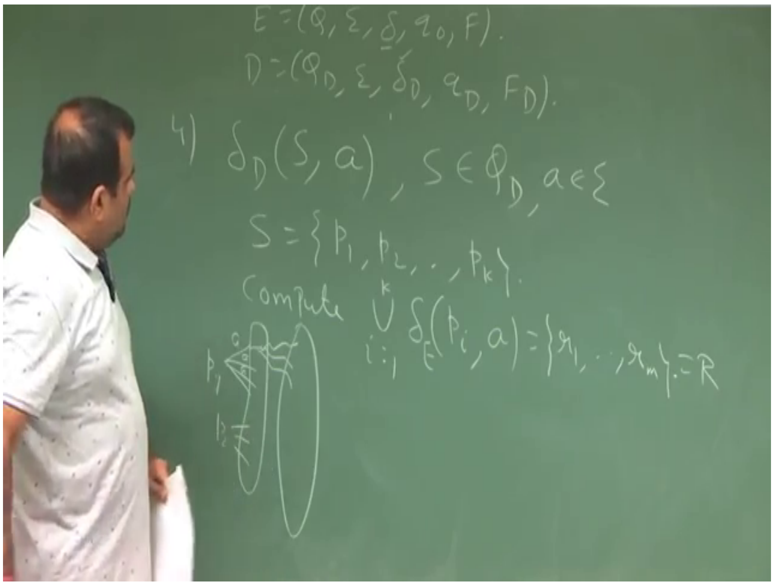
So, from q 0 what are the state we can move, we can go with the epsilon move only with no input reading. So, that is our epsilon closure of q 0 that will be our starting state ok. So, now what is the and also final state we can consider as F D.

(Refer Slide Time: 19:27)



We will talk about this after defining this F D is nothing, but so set of all states I mean these contain the epsilon closure such that S linear F is non empty that is the way we define nah. So, we were considering the subsets which is again considered the epsilon closure. So, those subsets we take, among those subset which is having a having one, at least one final state of e then that is the F D. So, those subsets we consider as a F D. Now, we define the rule the delta ok; now we define the delta. So, in the delta we will consider the, we will flag the epsilon move.

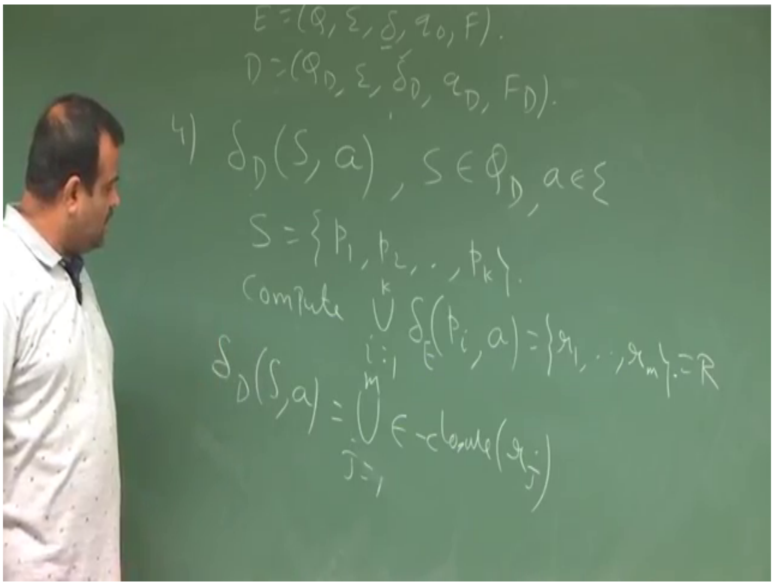
(Refer Slide Time: 20:27)



So, let us define delta D. So, delta D of S comma a, a is the input. So, this you have to defined where s is belongs to Q D, again S involve the epsilon closure and a is a is a alphabet is defined by. Suppose S is say some p 1, p 2, p k after doing this epsilon goes are everything we take the this one.

Then we compute, then we compute the p i, a. With this delta E and this is say i is equal to 1 to k and suppose this set is reaching to r 1, r 2, r m. So, we take p i. So, we take p i, from p i it is going to some set again we take p 1, we take p 2 like this so this is a, with a. So, this set is our this r set; now from this r set you have to now consider the epsilon closure of each of these and that will be our delta of this. So, that will be our delta of this. So, how to write that?

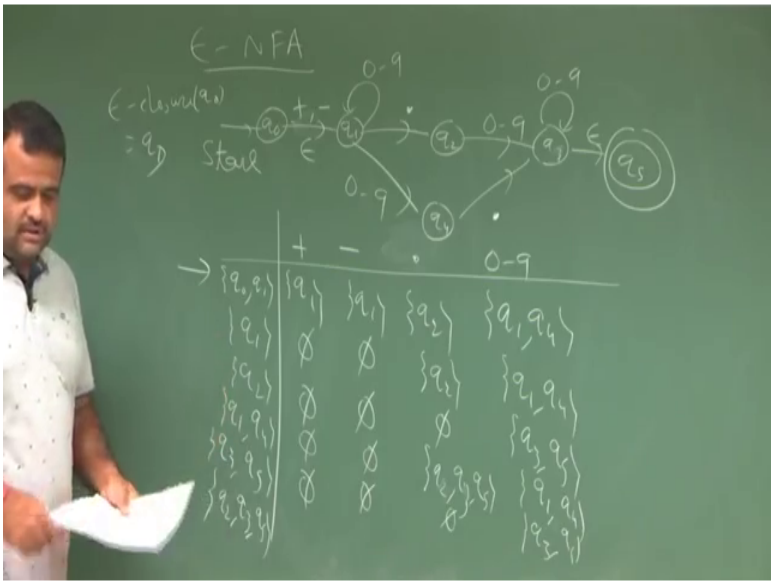
(Refer Slide Time: 22:21)



So, this is we compute this then delta of delta d of s comma a is nothing, but union of epsilon closure of this r i or r j j is equal to 1 to m ok. So, we take this p i's, from this p i we consider the move on a we go to some of the states. Now from there we consider the epsilon closure higher from we can move go by the epsilon move. So, we are capturing this epsilon move inside this. So, this is our delta of S comma a ok. So, this is this is the way we define that delta of S comma a.

So, now let us take a example and F, F we have defined. So, this is the way we define delta ok. Now let us construct that example. So, basically we are taking care of epsilon move in the state itself.

(Refer Slide Time: 23:40)



So, we take the same example like we did not the beginning, but here we will try to construct the DFA directly without helping of the middle NFA. So, this is the given epsilon NFA we have q 0 let me just draw it again q 1 q 2 q 4 or q 3 sorry q 3 and q 4 is somewhere here. So, we have a move plus minus and epsilon this is the starting state and from here we have a move and with dot we can go here again with an integer we can go here. With an integer we can go here and with dot we can go there and then with the integer we can and then we will go to the final state q 5. So, this is our epsilon. So, this is our epsilon.

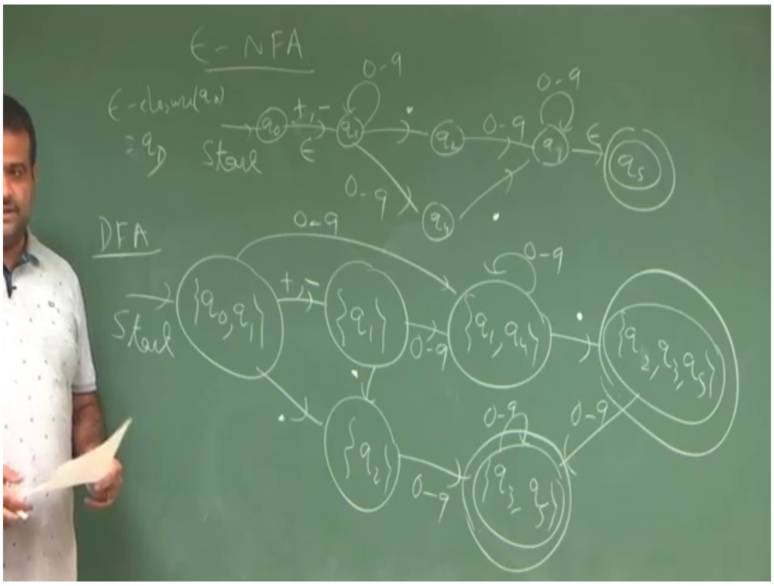
So, now from there we you want to compute that equivalent DFA. So, for the DFA we consider all the states which are which are just yeah we consider all the states which are including the epsilon closure of this and we find the delta. So, if you do that. So, first the starting state is nothing, but epsilon closure of q 0 that is our Q D. So, epsilon closure of q 0 is q 0 and q 1. So, q 0 and q 1 this is the starting state.

And then we consider all the state including their epsilon closure. So, q 1 q 2 then q 1, q 4 and then q 1, q 3, q 5 q 5 and then q 1 q 2 q 3 q 5 q 2 q 3 q 5. So, these are the states of this DFA and these are the epsilon closure also. So, plus dot dot 0 to 9 any input alphabets. So, with the plus we can just check this will go to q 1 again and this will go to q 1 and this will go to q 2 because if we are at q 1 q 2, if we have a dot we can take epsilon move to q 1 from yeah. So, from q 1 we can go to q 2 only and this is q 1 q 4 ok.

So, similarly this is empty this empty is also a state because empty is epsilon closure of nothing so epsilon. So, that is this is the q 2 you can verify this and this will be q 1 q 4. So, this way we will continue so this will be empty, empty and this will be again empty and this will be q 3 q 5 q 3 q 5 because from q 2 if we have a digits, from q 2 where we can go? From q 2 we can go to q 3 or from q 3 again by epsilon move we can go to q 5; so, q 3 q q 5.

So, similarly this will be this will be empty and this will be q q 2 q 3 q 5 sorry q 3 q 4, q 5 sorry q 2, q 3, q 5; q 2, q 3, q 5 and this will be q 1, q 4; q 1, q 4 you can verify this. And similarly this will be empty, empty, empty, empty and this is q 3 q 5 and last one you can verify this. So, if you draw this picture in a graph to will be like this.

(Refer Slide Time: 28:39)



So, let me draw the graph yeah. So, this will be like this. So, we have this is the starting state q 1 q 2 q 0 q 1 and these are the other state q 1, q 4 and then q 1 sorry q 2 q 3, q 5 and then we have a q 2 over here and then we have q 3, q 5. So, these are the possible state for the DFA and this is coming from the epsilon closure.

So, we if we have a plus dot we go to this state if we have a integer 0 to 9 any integer between 0 to 9 will go to this state and if you have a dot we go to this state. And from here if we have a dot we come here and from here again 0 to 9 any digit will go there and if we have 0 to 9 from here. So, this is we can easily verify with dot, now from here 0 to 9 any input and from here 0 to 9 and here we have a self loop with 0 to 9 ok.

And now who are the final state? Final state consists of so, here final state is q 5. So, wherever the q 5 content that will be the final state. So, these two are the final state. So, this is the corresponding DFA this is a variant of the substrate construction, but directly we are capturing the epsilon effect by this way ok. So, this is the example are from epsilon NFA we can directly construct the DFA. So, details will be given in the lecture not.

Thank you.