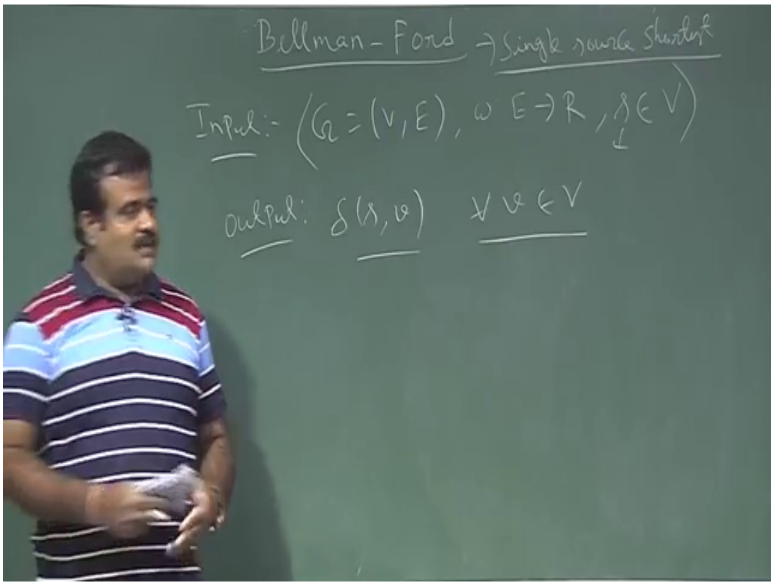
An Introduction to Algorithms
Prof. Sourav Mukhopadhyay
Department of Mathematics
Indian Institute of Technology Kharagpur

Lecture - 45
Bellman Ford

We talk about Bellman-Ford algorithm which is a general algorithm. So, it is not like Dijktras algorithm that there is a restriction on the edge weight, because Dijktras cannot handle the negative cycle. So, for that we had taken the edge weight to be non negative, but Bellman-Ford there is no such restriction. So, what are the input?

(Refer Slide Time: 00:44)

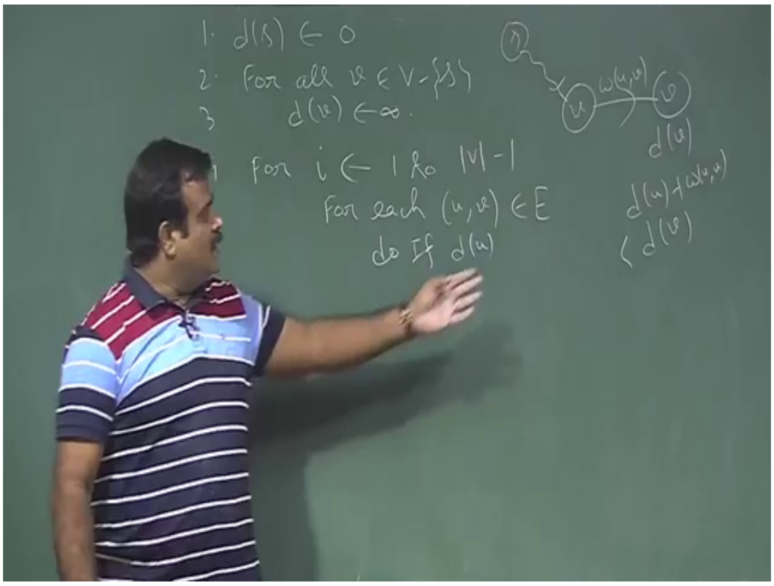


Input is basically a graph directed graph V comma E and the edge weight, so this is basically w to R. So, here there is no restriction of the non negative edge. So, edge could be negative also; that means, there is a chance of negative cycle. But any way Dijktras can handle Bellman-Ford can handle that. And there is a source vertex which is also another input. So, this is also a single source shortest path algorithm.

So, given a source we are finding the shortest path from that source to other vertexes. So, output would be delta of s comma v for all v belongs to this- either it is exists otherwise it will be infinity. So, otherwise the there will be a negative cycle we have to be put the negative cycle. So, this is the Bellman-Ford input output otherwise negative cycle will be there, we have to report that the Bellman-Ford should able to report that, because there is a chance of negative cycle also, ok

So, let us write the pseudo code for Bellman-Ford algorithm.

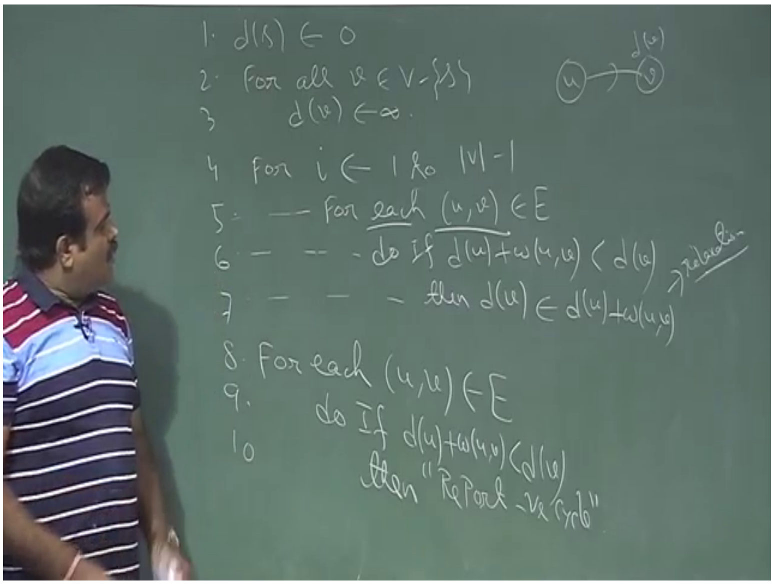
(Refer Slide Time: 02:23)



So, it is a basically the initialization part is same as Dijktras algorithm. For all vertexes so degree of v to be infinity initialization phase is same as Dijktras. So, this is the initialization. And then what we are doing we are doing branch of relaxation step. So, how many times? For i is equal to 1 to v minus 1 time. Why v minus 1 time we will come to that. Then for each vertex, sorry for each edge we will try to do the relaxation. We check weather do if d v; we take vertex u to v this is the vertex u to v sorry this is the edge u to v.

Now, this was having a degree; now we have somewhere s is there. Now this is a path form u s to v like this we go from s to v then u to v. Now if this path is better than this then you have to relax, so this is the relaxation step. If this path is the s to v we go to s to v and then u to v this. Now weight of that path is d u plus this, because d u is the distain estimate from s to that vertex. And if that is better than the degree the distain estimate what we have then we must relax that edge. So, that is the relaxation step we must do, ok.

(Refer Slide Time: 04:34)



So, if d u is greater than; sorry if d v or d e if d u plus delta u v is less than d v then we must relax this- then d v must be relax. So, this is the 5, 6, 7. So, this is the relaxation step. So, we are relaxing this vertex v. So, this is a direct edge u to v, so this is d v. Now if this d v is greater than this new path we come from s to u then this direct edge then we must relax this edge. So, this called this is the relaxation. And this we are doing how many times, this we are doing for all edges. So, this is the inner loop; when outer loop how many times v minus 1 time; why v minus 1 time will come to the correctness of this algorithm. So, there will see why it is v minus 1 time.

So, anyway this is the outer loop, this outer loop is we are running from v is n n minus 1 times or number of n times and the inner lop we are running form for each of this vertexes; for each of this edges. So, somehow we need to maintain the edge numbering. So, that we can just; we will come to an example of Bellman-Ford how it is executed. Now, this is the stage.

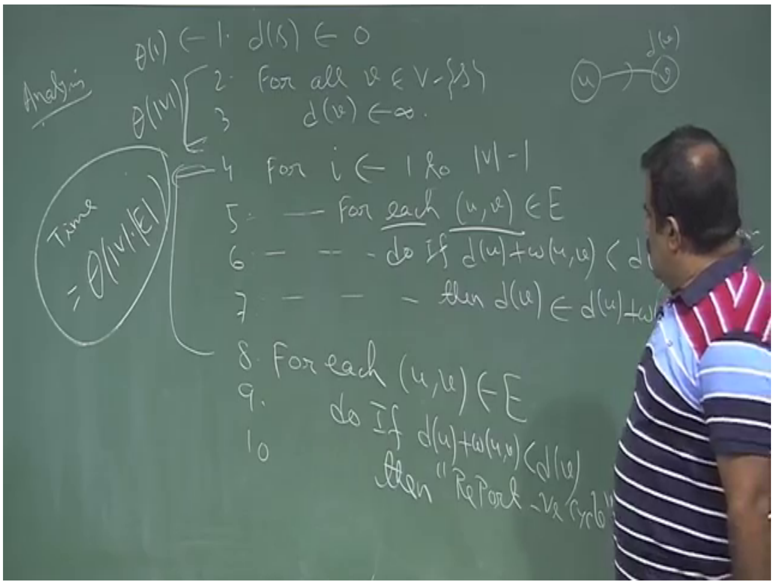
Now we have done, so after this step we have done in n number of relaxation. Now, still if we could further relax a edge then there is some indication that there will be a negative cycle. So, that step we have to follow. Otherwise after completion of 7 if there is no negative cycle then it must converge. That delta, that d v of this much converge to a delta of s comma v, but if after doing these so many relaxation we are doing this is v times e times. So, order of v into e times relaxation we are doing. So, even doing so much relaxation if even though we can reduce the a degree further; that means, even after this step if we can relax a vertex further then we must say that there is a negative cycle. So, it will never converge. So, it will keep on decrease.

If there is a negative cycle; that means, it will keep on decrease. So, that step you have to follow. So, this is the last step for each vertex; last time. If it is there is no negative cycle then there is no meaning of this step, but anyway we have we have this step to in order to have this checking of negative cycle.

Now, for each vertex this is last time, this is the final checking. So, if the d u is d u plus w u v is less than d v; that means, d v is a candidate for relaxation d. That means, there is a negative cycle. That is why making this; that is it is not converging. Then so report negative cycle; that is it. So, this is the pseudo code for Bellman-Ford algorithm. We just report the negative cycle, because we have done enough relaxation after that we just this is for the check of negative cycle; checking of negative cycle after that we just try one more time for the edge wise.

Now, then if we see further could be can be relax; that means, it will never converge again it will can be relax like this. That means there is a negative cycle. And in the correctness we will see why we are doing this loop for v minus 1 times, so that we will see in the correctness of this theorem. Now before going to the example let us talk about time complexity of this theorem. What is the time complexity?

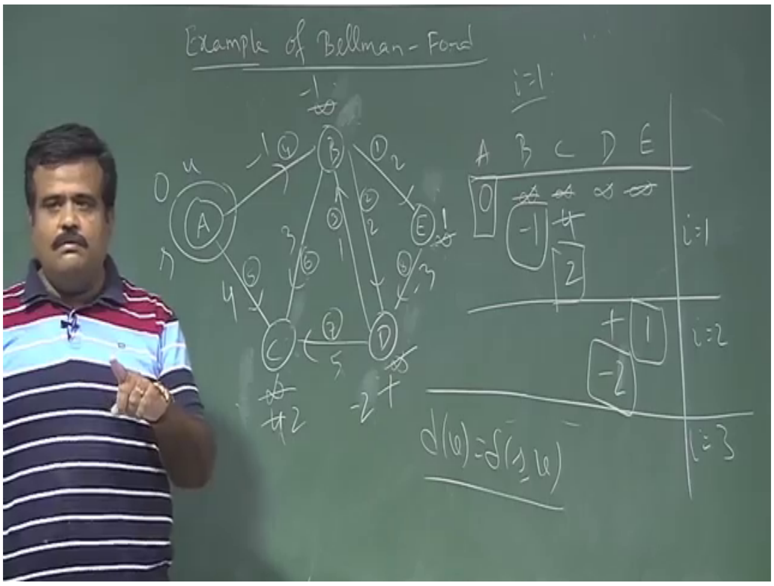
(Refer Slide Time: 09:41)



So analysis says time analysis. So, this initialization step is basically same order of v this is basically s; order of v. Now this loop is basically v times and this loop is e times. So, this is basically order of V comma E. So, this is the time for Bellman-Ford algorithm order of V times E, ok.

Now, let us take an example how this algorithm is working. So, let us take a graph then will execute this Bellman-Ford on that graph.

(Refer Slide Time: 10:42)



Example of Bellman-Ford: and at the end Bellman-Ford should give us delta of v is basically d v is basically delta of s comma v at the end of Bellman-Ford; otherwise it should be report the negative cycle. So, this is at the end of Bellman-Ford it should give us this. Anyway let us take an example. So, let us take a graph, directed graph A B C D E. So, let us have the edges. So, this is here, this is here, this is here, this is here. So, let us have this weight so minus 1, so here we have allowing the negative weight because it can handle the negative cycle: 5, 2, 2, say minus 3. So, minus 1, 4, 3, then 5, 1, 2, minus 3, 2 ok

So, this is the input of the Bellman-Ford algorithm, now we take a source vertex. Suppose s is our source vertex say. Now we put everything into the A B C D E. So, this is degree we are assigning 0 and each is infinity. So, this is initialization step. So, source vertex has degree 0 and each are infinity. Now we have to run this for we have a outer loop from i is equal to 1 to v minus 1 times and in inner loop we have to run it for how many times for each vertex each edges. So, somehow if we can have a age numbering it will help us.

So, let us numbering the edges this is number one edge say this is this numbering is not unique you can have your own numbering and you can follow this numbering in each iteration, ok. This is number 3, may be this is number 4, this is number 5, this is number 6, this is number 7, this is number 8 edge we are just numbering this edge in a in order to do it for all the edges because if you recall that pseudo code for all i we have to do it for all the edges. Now, for i s equal to 1 we have to do it for all the vertex. So, we initialize the degree this is 0, this all are infinity, all are infinity, except this. Now we start with this vertex this, this is u this is v now both are infinity. So, no relaxation is possible. So, there is nothing is change there for age number one. Edge number two also this is infinity this is infinity. So, there is no relaxation is possible for this vertex. Edge number 3 same no relaxation. Edge number 4 is this now for edge number 4 this is u this is v. Now d v is infinity: now we have a path which is d u 0 plus minus 1. So, this is minus 1.

So, this is u this is v. So, this was infinity, now we have somewhere s is there, so this is direct path. So, this is minus 1. So, this is degree is 0 so this minus 1. So, this must relax because we have a better path. So, this edge must relax to minus 1. So, this must relax to minus 1. So, which is b b must relax to minus 1. So, this is edge number phone number 4 now edge number 5 also similarly this must relax to 4. So, this is c, so this must relax to 4. And then 5 then 6; 6 now if you look at 6 this is u this is v. So, u v this is u this is v. Now the degree of this is basically degree of this is v 4, now degree of u is what degree of u is basically minus 1 and we have a direct path with 3.

So, s is somewhere here that means, we have a path form a s to this node with minus 1 then we can take this direct path. So, that is a new path from s to v with a cost minus 1 plus 3, it is basically plus 2 which is greater then 4; 4 must relax this. So you must relax this 4 to b 2, ok because that is the better part. So, this is basically c is now became 2. So, this is the relaxation we are doing. So, this is edge number 6. Now edge number 7 edge number 7 this is infinity. So, this will be no change edge number 8 also no change, because this is infinity this is infinity there is no change. So, edge number 2. So, all edge exhausts. So, this is for i is equal to 1.

Now, for i is equal to 2 if you remember the outer loop is for i is equal to 1 to v minus 1; I mean number of edge minus 1 number of vertex minus 1. So, now, is equal to 2 i is equal to 2 will we have to again we have to do the inner loop for all the edges. Now since we have already have a edge numbering we can follow this numbering or we can have a different numbering, but we have to make sure that we have check this for all the edges that inner loop is for all the edges. But if we do the numbering, so if we follow the numbering there will be a simplicity in this algorithm.

So, we are basically following the same algorithm. But it is not required to follow the same numbering for the next loop, we can have a different numbering also. Only thing we are ensure that we are doing this for each edges, so that we have to ensure, ok. So, if we do this for i is equal to 2 we do we are following same thing, so we start with this vertex yeah sorry this edge this is u, this is v, this is u, this is v, now this is having degree infinity. Now we have a new path which is basically form s to v u we can go minus 1 then we can take this two which is basically minus 1 plus 2 1 which is better than infinity. So, this will be now 1. So, basically E is now became 1; E is now 1.

So now, edge number two is this one this is u this is v again this is minus 1 this is plus 2 so this will again become 1. So, d will become 1. So, this is edge number 2 edge number 3 is this one. Now, this is 1 that means, there is a path from s to this node this is u this is v. So, there is a path from s to this node with a cost 1 and we have a direct edge we pay cost 2. That means, now we have a path from s to this node with the cost 1 plus 1 which is basically 2, but we already have a path which is minus 1 which is not better. So, we do not have to relax this, this node is not getting relaxed because the path which you already have degree is better than the path which is currently here.

So, we are now relaxing that so come to 4. So, this is edge number 3, edge number 4 if you come to edge number 4 this no change edge number 5; edge number 5 is here. So, this is 2 this is 4 so no change no relaxation is possible, no relaxation because this is better path we have now edge number 6 edge number 6 is this 1. So, this is basically what this is u this is v for this edge 6. Now this is this is d v is minus 1 then if we take this path 3. So, this is minus 1 plus 3 2. So, we have a path 2 to form s to this node, but we have already path 2. So, there is no change. Now, we are not relaxing this edge. So, edge number 7 edge number 7 this is u this is v. So, this is u means, so is the degree of u that means, you have a path from s to u with a cost 1 and then we can take this direct edge with the cost 5. So, that will give us a path from s to this node with the cost one plus 5 6, but which is not better than 2. So, we are not relaxing this vertex v.

So, then edge number 8: for edge number 8 this is u this is v, you have to check whether we can relax this v. So, the degree of v is now what degree of v is 1, now what is the degree of u degree of u is basically 1. That means, we have a path from s to this with a cost one then what is the direct edge it is minus 3. So now, we have a path from s to this by this we take this d u 1, then we take the direct edge minus 3, so this is basically 1 minus 3 so minus 2. So, minus 2 is better than this. So, you must relax this. So, this will be now minus 2. So, v is the v is relax. So, d is now minus sorry minus 2. So, this is edge number 8. So, this is for i is equal to 2.

Now, for i is equal to 3 we need to do because this vertex how many vertexes one two 3 4 5. So, 5 vertexes means, so the outer loop is v minus 1 times that means, 5 minus 1 at least 4 times we have to do this. Now i is equal to 4 you have to do. So, for i is equal to 3 we can just do it again. So, again we have to execute the inner loop for all the edges. So, we have to just check the all the edges and you have to see whether we can relax the vertex. So, edge means u to v. So, we have to check whether the v vertex could be relax, ok.

So, now which start with the; we can follow the same edge numbering or you can have a different edge numbering, it does not matter. So, since we already have a numbering we can just for the simplicity we can just follow the same numbering. So, we start with this edge number 1. So, if you start with edge number 1 this is u this is v, now this is minus 1 this is 2. So, this is plus 1, so no change same. So, no further relaxation is needed for this vertex v. Now edge number 2 this is u this is v. So, this is basically minus 1 and plus 2. So, this is basically plus 1. So, this is already having minus 1. So, plus 1 is not better than minus 2. So, no relaxation is required for this vertex, ok.

So, edge number 3: edge number 3 is this one this is u this is v. So, this is minus 2 plus 1, so it will be minus 1. So, it is same as which is the same the degree which is having. So, there is no relaxation is needed for this vertex. Now number 4: number 4 is this one. So, this is basically minus 1 which is same. So, no relaxation is required for this vertex v. So number 5: number 5 is with one, so this is 0 plus 4 is 4 which is not good as 2. So, no relaxation is required for this vertex. Then 6 number: 6 vertex is this one, so this is minus

1. So, minus 1 plus 3 is 2. So, this is already same as this so there is no further relaxation is required for this vertex v. Then number 7 is this one; so minus 2 plus 5 so 3. So, 3 is not good as 3 is not better than 2. So, there is no further relaxation is required for this vertex v. So number 8: number 8 is basically this one u v. So, this is mine, this is one this is minus 3 so minus 2 which is same as this so no further relaxation is required, ok.

So, this is converging. So, this converge this d s is become converge. So, there is no further changing happening for i if we just have another loop. So, this is converging with the; this degree is converging with the delta. So, delta v is now be become s of s comma v. So, these are basically the weight. So, these are basically our deltas. So, that is basically converging, because there is no negative cycle in this graph. If there is negative cycle, I will suggest you to do an example in the home. So, you just put a negative cycle here may be some; so this is. So, this is a maybe you take this one this, this and this. So, try to put a negative cycle over here just put a minus 2 over here then there is a negative cycle then try to execute this. Then it will not converge you can just verify this it will not converge it will keep on relaxing the edge; keep on relaxing the vertices over in this region. So, it will not converge like this. So, if it is converging; that we that is why we have a final check.

And also in the correctness we will we will discuss why this outer loop is for v minus 1 time. So, this we will discuss in the next class.

Thank you.