Principles of Communication Systems - Part II

Prof. Adithya K. Jagannathan
Department of Electrical Engineering
Indian Institute of Technology, Kanpur

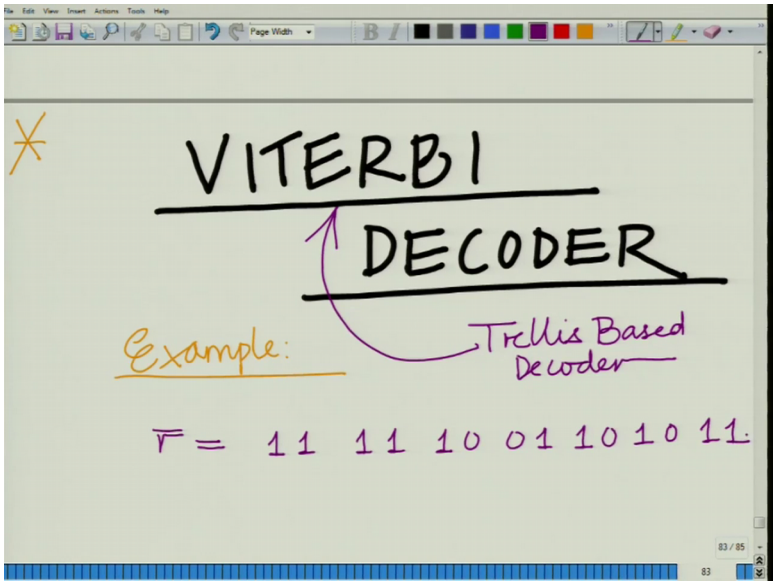
Lecture - 56

Viterbi Decoder for Maximum Likelihood Decoding of Convolutional Code Using Trellis Representation, Branch Metric Calculation, State Metric Calculation, Example

Hello. Welcome to another module in this massive open online course. So, we have looked at the principles of decoding convolutional code.

And now let us look at how to actually apply these principles and the decoding of the received word in a convolutional code. And this decoder which finds the maximum likelihood estimate of the transmitted code word is known as the Viterbi decoder which is a very very important and celebrated algorithm in communication which has made; you can say modern communication of modern communication possible at a very low error right.

(Refer Slide Time: 00:54)

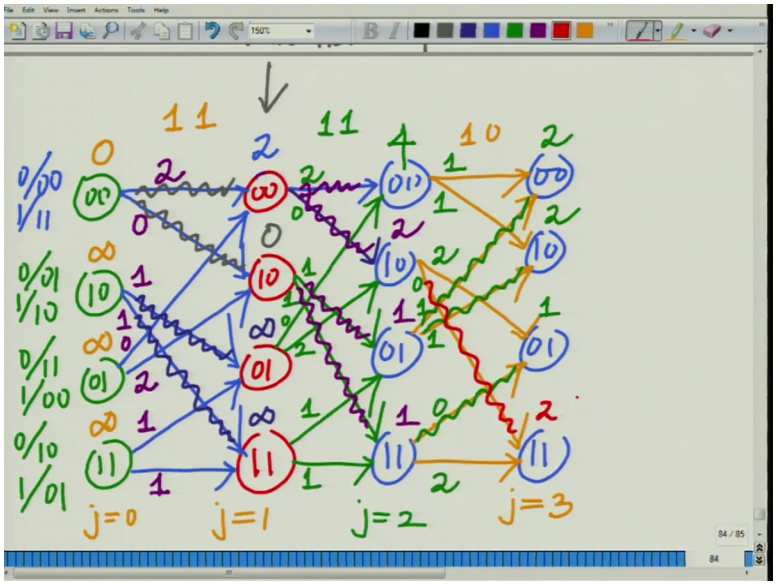


So, what we are going to look at is the Viterbi decoder for a convolutional code, which was proposed by an Viterbi. And we are going to illustrate this trough an example. So, we are going to consider an example. And of course, was without saying this is a trellis base decoder that is by we have been building up and emphasizing the importance of the

trellis. This is the trellis based decoder. Let us look at the received sequence r bar equals 11, 11, 10, 01, 10, 10, 11: 0 01 10 10 11.

Now, corresponding to this let me begin by drawing the trellis diagram. So, I will begin by drawing trellis diagram corresponding to this.

(Refer Slide Time: 02:16)



So, what I have over here is basically the first stage of the trellis, let me start by drawing the first stage of the trellis diagram. So, this is our states 00, 10, 01 and 11. And we know at the next time instant we have a similar repeatitions of the states, so at the next time instant we have again 00, 10, 01, 11.

And now we show the transitions, remember there are each state splits into two arrows. So, this is the top arrow, there is a bottom arrow, top arrow, is a bottom arrow, top and this structure is repeated. Top arrow denotes 0 output 00 input 1 output 11; state 10 0, outputs 01 the bottom arrow denotes input 1; state 10 01 input 0 outputs 11 input 1 outputs 00; and state 11 input 0 denotes output 10 input 1 denotes output 01 corresponding to the bottom arrow.

And now we see that the received code word corresponding to this is 11. So, I am going to write it here at the top of this and the top of this trellis I am going to write received code words. So, this is corresponding to the states at j equals to 0, states j equals to 1 and the received code word is 11.

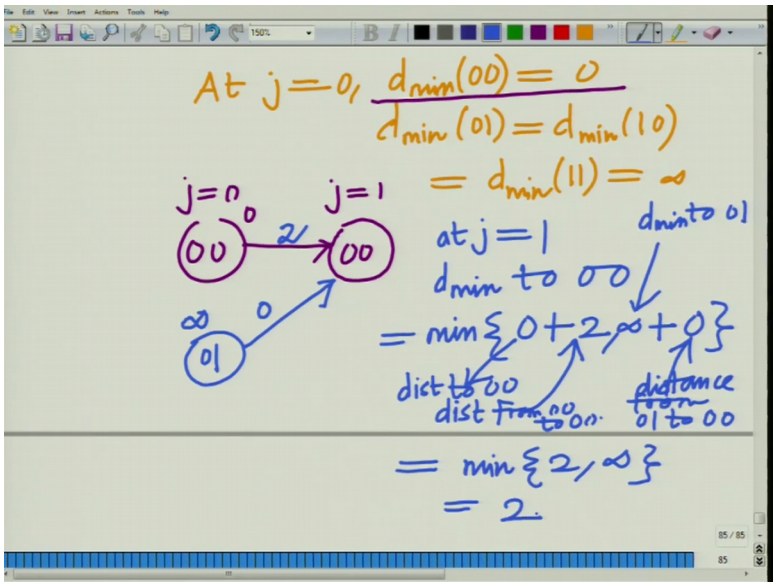
Now you see at time j equal to 0 we assume that we are starting from the state 00. So, I am going to set the d min on each state I am going to write the d min. So, d min to 00 the state 00 is 0. For all the other states I am going to set the d min as infinity. So, at j equal to 0, so this is an important point at j equal to 0 d min of the state 00 equal 0 and d min of any other state; or sub for instance let me right it express it d min of 01 equals d min of 10 equals d min of 11 equals infinity. Just to show that these states we do not begin with any way begin at the state 00. So, there is a distance of start with d min of state 00 equal to 0.

Now if you look at this; now let us look at the branch between 0 and 0 to 00: a state 00 to state 00 that corresponds to the output 00. So, the hamming distance between the received code word; remember, the total metric is the sum of the branch metrics. The branch metrics is determine by two things: one the received word corresponding to that time instance which is 11, and to the output corresponding to that branch- output corresponding to this branch is 00. So, the hamming distance between 00 and 11 is 2. So, the branch metrics corresponding to this branch is 2.

Similarly branch metric corresponding to this branch which is output 11 received word is 11 so the hamming distance is 0. Similarly, branch metric corresponding if you look at 10, the top branch is 01 output 01 received word is 11. So, hamming distance is 1. Similarly the bottom branch hamming distance is 1. Top branch 01 output 11, so hamming distance is 0. Bottom branch output 00, so hamming distance is 1. 11 top branch output is 10, so hamming distance from 11 is 1. Bottom branch output the coded outputs are 0, the coded output in bits are 01. Hamming distance between 01 11 is 1, ok.

Now, remember we use principle number 2. Now look at this at stage 00 a time instant at j equals to 1 I can either arrive from 00 or I can arrive from 01. So, if you right this separately.

(Refer Slide Time: 07:38)

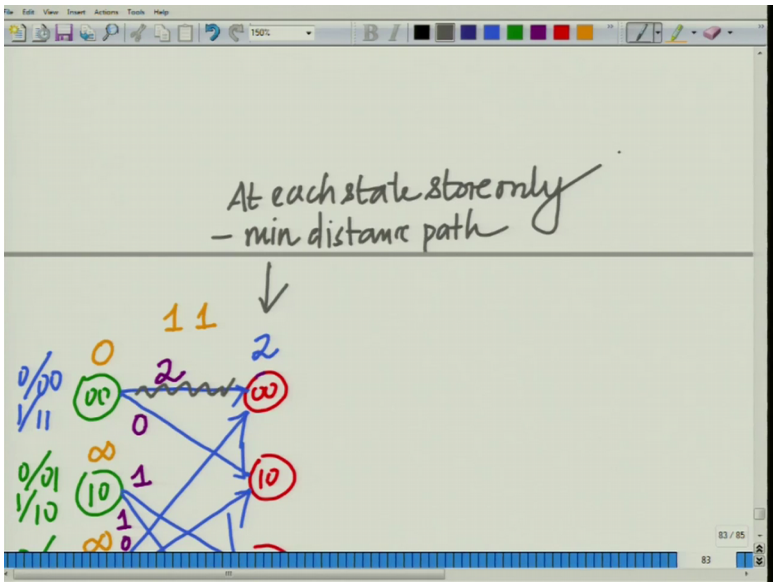


Look at this: I have stage 00 at j equal to 1. Now I can either arrive from 00, the corresponding minimum distance to 00 is 0 and the branch metric remember is 2; which is the distance between 00 to 0. Or I can also arrive from the state 01, if you look at it I can arrive from the state 01 also to 00. The corresponding branch metric if you look at this, the corresponding branch metric is 0. The corresponding branch metric is a top branch it is 0, but the minimum distance is 01 is infinity; remember be initialized is a infinity.

Therefore, now at j equal to 1 d min to 00 equals minimum of well distance to 00, that is 0 plus distance from 00 to 0 2 and infinity plus 0. That is distance to 01 and distance between 01 and 0. So, this is distance to 00, this is distance from 00 to 00, this is d min to 01, and this is distance from 01 to 00. And now you can see this is equal to minimum of 2 comma infinity which is naturally equal to 2. So, the minimum distance is 2

And the minimum distance branch is this. Remember at each state I need to store only the minimum distance; remember that is our principle number 1 minimum distance path and what is the distance. So, at each state we store only the minimum distance path. At each state we store only the minimum distance path.

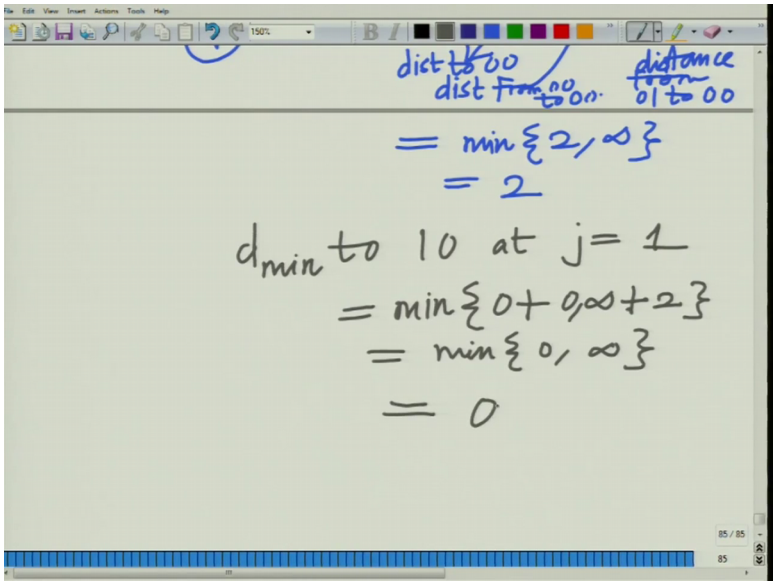
(Refer Slide Time: 10:14)



So, to 00 at time instance j equal to 1 the minimum distance is 2 and minimum distance path is from 00 take the top branch with output 00 and branch metric 2.

Now similarly if you look at the state 10; now 10 I come to 10 from 00 or I can come again I can end up in 10 from 01 from at time instant j equals to 0. So, minimum distance to 10 again is minimum of the distance to 00 that 0 plus branch metric is 0 or infinity that is the distance minimum to 01, minimum distances to 01 plus the branch metric between 01 and 10 that is 2.

(Refer Slide Time: 11:35)



So, the minimum distance now you can see d min to 01 or to 10 at time j equal to: if this is equal to I can come from 00 d min to 00 0 plus 0 the branch metric or I can come from 01 that d min to 01 is infinity plus the corresponding branch metric between 01 and 10 is 2, which is equal to minimum of 0 comma infinity which is clearly equal to 0. And therefore, the d min to 10 is 0 and the minimum distance is a path is this.

Now here if you look at 01. Now if you look at 01, 01 I can come either from 10 which has d min infinity. So, the minimum distance to the 01 is infinity plus the branch metric 1 or I can come from 11 again which has a d min infinity so the distance is again infinity plus 1. So, the minimum distance is minimum of infinity plus 1 comma infinity plus 1 which is any case is infinity. So, it does not matter I can take whichever path. Again similarly the minimum distance to 11 is also you can see infinity, because I can either come from 10 or 11 and both of them have the minimum distance of m.

And you can see that at every state at each time instant for every state I am noting two things: one is the minimum distance to that and the branch corresponding to that minimum distance because we are going to need that later, because when you find minimum distance path that is the path through this trellis which corresponds to the minimum distance metric; minimum hamming distance with respect to the received word I am going to need these minimum distances and the corresponding path.

And now we can repeat this at every subsequent time instant. For instance, again I drop the next stage of trellis this corresponds to the received word 11 again at the time instant j equal to 1 received word is 11. Once again the minimum the branch metrics will be the same 2 0, 1 1, 0 2, 1 1 minimum distance to 0 is or I can come at j equal to 2 I can come to 00 from 00 so minimum distance is 2 from 00 plus the branch metric or I can come from 01 from minimum distance is infinity plus 0 from 01. So, the net minimum distance to 00 is minimum of 2 plus 2 or infinity plus 0 which is basically minimum of 4 comma infinity so the minimum distance is 4.

For 10 I and the minimum distance of path, do not forget to mark the minimum distance path that is the branch between the 00 and 00. Now, 10 I can come from 00 the distance is 2 plus 0 or I can come from 01 distance plus is infinity plus 2. So, needless to say the minimum distance is 2 plus 0 from 00 so that is 2 and this is the minimum distance branch.

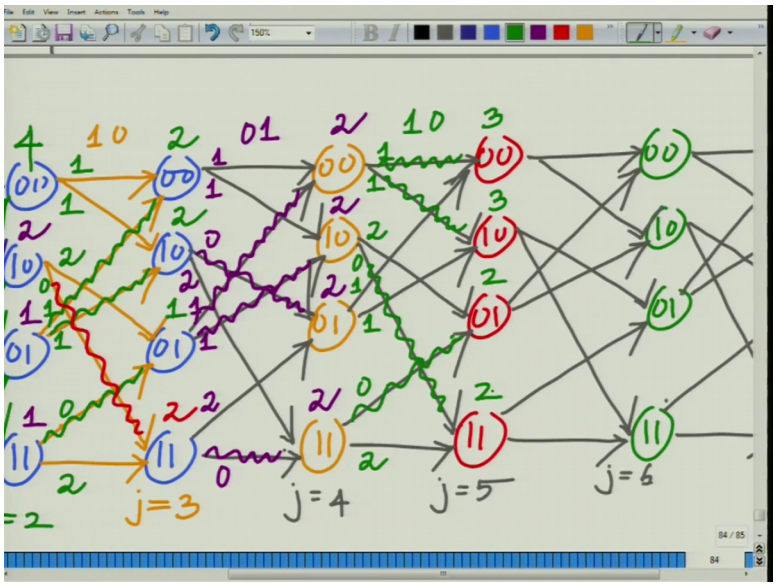
01 I can come from 10 from 10 right or I can come from 11, from 11 as minimum distance of infinity so its infinity plus 1, from 10 it is 0 plus 1 needless to say the minimum is 0 plus 1 that is a minimum distance of 1 close to 01 to 11 I can come either from 10 the minimum distance is 0 plus 1 the branch metric or it is infinity plus 1 from 11 needless to say the minimum is 0 plus 1 that is 1 and that comes from 10. This is at j equal to 2.

Similarly, I can do j equal to 3 which is 00 10 01 11. I have all the branches we can keep doing this j equals to 3. The received code word corresponding this is 10, the branch metric will change the branch metric the top branch 00 to 00 that as an out the output is 00. So, hamming distance between 00 and 10 that is going to be 1. Hamming distance bottom branch output is 11 hamming distance is 11 and 10 is 1, so hamming distances are 11. Hamming distance now 10 corresponding to 10 and top branch will have hamming distance 2 corresponding to 01 bottom branch is 10, so hamming distance is 0 01 11 correspond to the hamming distances will be 1 and 1 again. From 11 there are two branches 10 that top branches output that will hamming distance 0 bottom branch is output 01 that will be have a hamming distance of 2.

And once again the minimum distance to 00 is either minimum of 4 plus 1 from it is 00 to 00; 4 plus 1 or it will be 1 plus 1 from 01 distance minimum minimum distance is 1 branch metric is 1. And needless to say minimum is minimum of 4 plus 1 and 1 plus 1, so the minimum is 1 plus 1 that is 2 at this comes from 01; this comes from and this is your minimum distance path. To 10 the minimum is either 1 plus 1 or 4 plus 1 once again it is 2. So, once again it is 2 and it comes from 01. Minimum distance to 01 is either 2 plus 2 from 10 or 1 plus 0 from 11. So, minimum is 1 plus 0 that is 1 and that comes from 11. And finally, between to 11 either you can take from 10 that is 2 plus 0 or from 11 that is 1 plus 2 the minimum is 2 plus 0 from 10, so that corresponds to minimum distance of 2.

So, that corresponds to j equal to 3. So, we have corresponds to j equal to 3.

(Refer Slide Time: 19:24)



So, I am going to quickly drop the rest of the stages of the trellis they save some space, because I am going to need it later brought two more stages; that is what we are going to need: 10 01 and you can see what I am doing is simply repeating all the states at each stage. So now I am going draw the branches. This is very simple because all you need to do is you have to repeat the trellis, you have to repeat the stages at every time instant. It is a repeated structure which makes it every convenient and advantages, I am just going to complete this example for the purpose of illustrating it.

And now you can see this corresponds to j (Refer Time: 21:38) j equal to 5, j equal to 6, j equal to 7; I am going to quickly keep writing down the received word 10 corresponding if you corresponding to the equal to 3 that will be 01. The branch metrics will once again be for the two branches from 00 11, branch metrics corresponding to 10 because it will be 01 top one is 0 output 01 so this as branch metric 0 bottom one will have a branch metric 2. 01 its 11 and 00 so again branch metrics will be 11, and here again 11 10, so branch metric is 2 and a lower branch will have 0.

The minimum is 2 plus 1 at j equal to 4 to 00 you can either come from 00. So, minimum is 2 plus 1 for 1 plus 1 from 01, so needless to say is the minimum will be well 2, at the minimum branch will be this. To 10 the minimum will be either from 00, so it will be minimum of 2 plus 1 or minimum of 1 plus 1 from 01. So, naturally the minimum is 1 plus 1, so this will be minimum of 2. And to 01 I can either come from 10 so 2 plus 0

that is 2 or from 11 2 plus 2 that is 4. So, the minimum is naturally 2 plus 0; that is 2. At the minimum distance branch is between 10 and 01 and to 11 I can either come from 11 that is 2 plus 0 2 or from 10 that is 2 plus 2 4 needless to say the minimum distance branch corresponds to this.

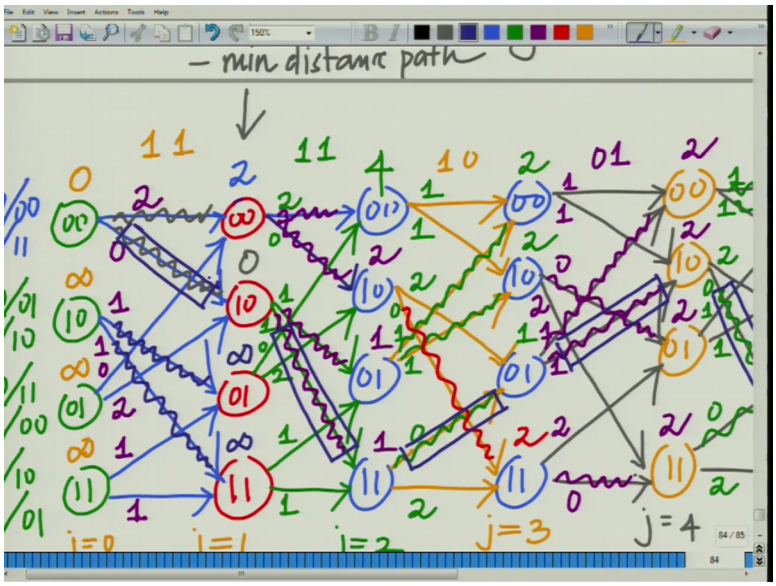
Now, look at state j equal to 5. The minimum distance between is to 00 will be either, so the branch metric now the corresponding received word is again. So, after 00 this is 10 the branch metrics will be 11 11 and 01, so this will be top branch will be 2 bottom branch will be 0. Between 01 top branch will be 1, bottom branch will be 1. And from 11 the top branch will be 0 bottom branch will be 2, because bottom branch corresponds to output 01 hamming distance between 01 and 10 is 2.

So, again if you look at to 00 I can come via 00 that is minimum distance will be 2 plus 1 that is 3 or from 01 minimum distance will again be 2 plus 3. Now here what we see is we have a tie, when we have a tie then obviously the minimum distance can correspond to anyone. So, we can choose any particular branch, it does not related.

So, in this particular case I can come to 00 from 00 the distance the minimum distance will be 2 plus 1 or I can come from 01 minimum will again be 2 plus 1. So, I can choose any particular branch. So, in this case I will choose the top branch. So, this will be 3; to 10 I can come again from 00 2 plus 1 that is 3 or I can come from 01 which is again 2 plus 1 3, so I can tick this which is 3. To 01 I can from 2 plus 2 2 plus from 11 that is 2 plus 0 2 plus 0 2 or I can come from 10 that will be 2 plus 2 4. So, needless say it will be 2 plus 0 that is 2. To 11 I can come from 11 that is 2 plus 2 4 or I can come from 10 that is 2 plus 0 2. So, needless to say it will be 2.

Again proceeding to j equal to 6; again j equal to 6 the received word is 10, j equal to 6 the received word is 10.

(Refer Slide Time: 26:15)



So, 01 let me just check 10 and the final one is 11, let me just write that as well 10 11. And again if you look at the branch metrics they will be the same. Whenever the code word received word is the same the branch metrics are going to be the same, right because the code word outputs for each branch are fixed. So, this is 11, 11, this is again going to be let me just write it with a different color.

This is 1 1 2 0 again 1 1 and 1 0; so 0 2. Minimum distance to 00 is either 3 plus 1 4 or 2 plus 1 3, so needless to say it is 2 plus 1 3 and minimum distance to 10 is either 2 plus 1 3 or 3 plus 1 is 4 and it is again 2 plus 1 3 from 01. To 01 it is either 3 plus 2 5 from 10 or 2 plus 0 that is 2 from 11, so needless to say it is 2. And for 11 it is either 3 plus 0 that is 3 2 plus 2 that is 4. So, needless to say that is 3. So, we have 33 to 3. And in the final stage the distance is the branch metrics are going to be; well the branch metric the top branch 00 to 00 that is 00. So, the hamming distance is 2 0, again 1 1 2 0 11 and this 1 from 10 1 the branches are 11 which will have a hamming distance 0 or a 00 which will have a hamming distance of 2 and the bottom branches will have a distance of 1 each.

And now once again you can find the minimum two distance to 00 is 3 plus 2 is 5 or 2 plus 0 2, so that will naturally be 2 plus 0 that is 2. Minimum distance to 10 is 3 plus 0 that is 3 or 2 plus 2 that is 4. So, naturally it will be 3 plus 0 that is 3. Minimum distance to 01 will be 3 plus 1 4 or 3 plus 1 4 you can choose either one, so that will be 3 plus 1 4. Finally, minimum distance to 01 I can come from 10 the minimum distance correspond

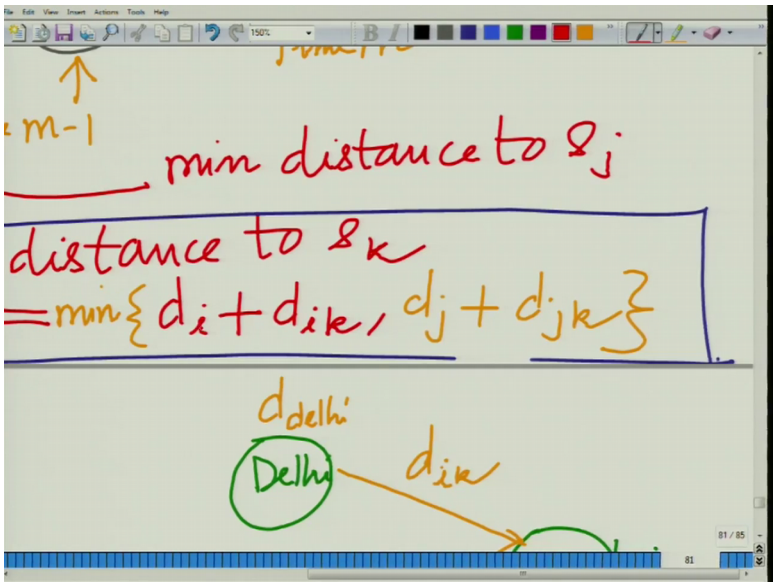
you direct is deeming to 10 3 plus the branch metric 1 4 or 3 plus 1 4 corresponding to coming from 1 I can choose either one. So, basically if I choose this branch this will be 4.

So now, what I have done is basically after this procedure; I understand it might little bit a confusing but argue to go were this again to understand it. Basically we have done nothing but use the principles which we have clearly explained or illustrated in the previous module. That is basically at every state, at every time instant I only need to store the minimum distance and minimum distance path that leads to that state at that time instance.

And to whenever you are trying to find the minimum distance to a state at a time instant from states the previous time instant, remember there are two states at the previous time instant which will lead to the state at the current time instant. So, I have to find the minimum distance from either state will be for instance let us say s 0 and s 2 are the minimum states that lead to s 2. So, it will be minimum of distance to s 0 plus branch metric between s 2 and s 0 and the minimum of s distance d min to s 1 plus the branch metric between s 1 and s 1. That is the principle that we have seen.

For instance, let me just go back and look at what we have done in that.

(Refer Slide Time: 30:58)



That is the minimum distance to s k from state i and j will be minimum of d i d min to

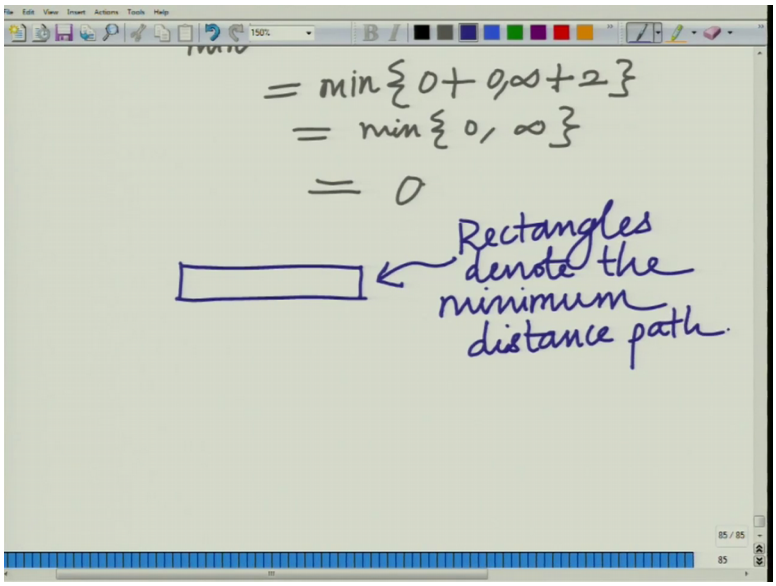
state i plus the distance between state i and state k comma d j minimum distance to state j plus d j k; that is the minimum distance that is the branch metric between state j and state k. So, that is basically what it is. And now you see at the final state we choose the one that is minimum distance. The one the that is the minimum distance is basically the 00.

And therefore, now we will trace back the path. At this point we will trace back the minimum distance path, because we have stored the minimum distance branch for every state- trace back. And to trace back the minimum distance path now you all have to do is trace back these basically the minimum distance branches. So, basically that gives me if I am going to put it in a rectangle.

So, the minimum distance come from 01, to 01 the minimum distance comes from 10 11, to 11 the minimum distance is this, to 10 the minimum distance is this, and to 01 the minimum distance is this through 11, and to 11 the minimum distance is through 10, at 10 the, wait there might be a slide apologize for this. And 11 the minimum distance is 10, and to 10 the minimum distance is from 00.

So, the state sequence will be now back trace, now you can draw the state sequence will be you can see 00 to 10 to 11 to 01 to 10 to 11 to 01 00. So, the minimum distance path is indicated by the rectangles we have back trace.

(Refer Slide Time: 33:46)

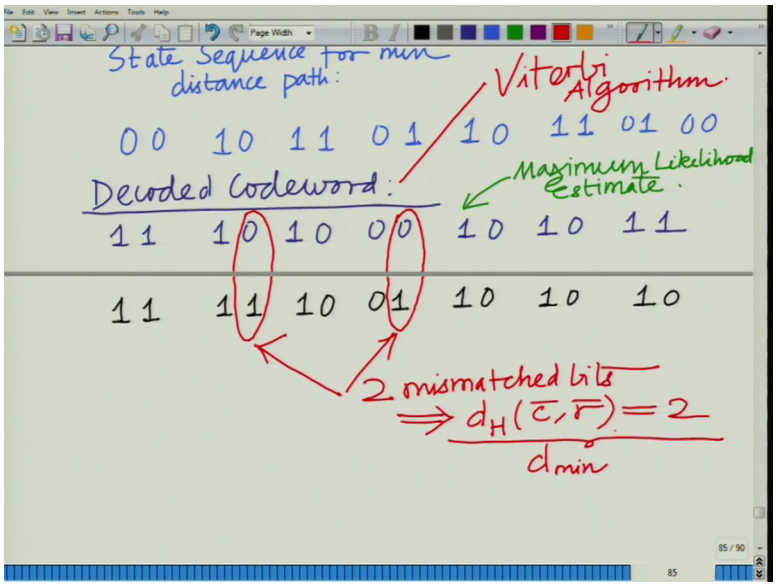


So, let me just mention it. These rectangles denote; we have back trace basically what we

have done at this point we have back trace to find the minimum distance path. And now the minimum distance the rectangles denote the minimum distance for and correspondingly leading that minimum distance that states along the minimum distant path I can find the state sequence.

So, the state sequence corresponding to minimum distance path or basically the maximum likelihood state sequence.

(Refer Slide Time: 34:31)



So, the state sequence for your minimum distance path now you can see this is basically; the state sequence is 00, 10, 11, 01, 10, 11, 01 and 00. This is the state sequence. Remember this is not the output code word. Now the decoded code word will become the output corresponding to this state sequence.

So, the decoded code word will be: remember this is the maximum likelihood decoded code word that will be 11. You can just look at it for instance corresponding to this branch it will be this is a bottom branch. So, 11 next is the bottom branch of 10, so that will be 10, next is the top branch of 11 that is basically 10 and so on. So, basically you will get the decoded word 10, 11, 10, 10, 00, 10, 10, 11. So, that will be 11, 10, 10, 00, 10, and 10. And now if you look at this with respect to the; and remember this is the maximum likelihood decoded code word. This is the maximum likelihood, it is not guaranteed; obviously.

Now, one you would understand it does not mean that this is the actual code word that has we transferred. We have to realize that this they can still be an error, this only denotes the maximum likelihood estimated that is what is the code word corresponding to received word as the maximum provability of having been transmitted. That is the as which one as the maximum likelihood of having been transmitted corresponding to the resume, that is all this says.

And now look at the receive code word, the received word is well let me just write this again 10, 01, 10, 10, 10, and what you can see is basically you have to the hamming distance between this and this basically you have two errors between the received word and your maximum likelihood. There are two places of mismatch

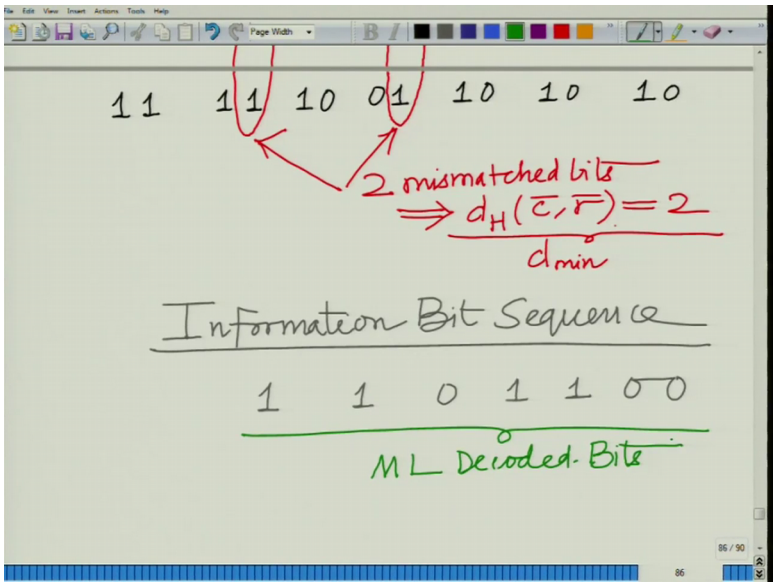
And that is what. In fact, we had seen when we completed the Viterbi algorithm the final. So, this two places of mismatch, two mismatch bits implies d H between c bar decoded code word an r equal to. And in fact, Viterbi algorithm guarantees that this is the d min and that is what we have seen. Remember from the final state this is your d min. There is a minimum hamming distance there is a code word corresponds to the path with a minimum path metric which also is the minimum hamming distance between decoded code word and receive word.

In fact, that is what is we are trying to do, by adding this branch metrics we are trying to find the path which corresponds to the minimum distance; minimum distance between received word and possible code word. The code word which as the minimum distance corresponding to the between the minimum hamming distance with respect to the received word. And therefore, the decoded code word, we use the Viterbi algorithm decode this, which is very important to remember as a emphasized most modern communication system digital communication systems employed one Viterbi algorithm in one form or the other that is Viterbi algorithm or different versions of it or one form of other. In fact, this was one of the earliest possible codes that were employed in wireless communication systems.

And which have made reliable wireless communication, possible even our noise channels. That is when the channels are very noisy which lead to error for bits and the receiver a reliable decoding. Correct, reliable transmission or such channels is possible through the. You remember the codes or the convolutional codes basically at channel

codes, which are used to recover that is basically both detect and correct errors that recovering over noisy channels. In fact, the Viterbi decoder is what made this low complexity decoding of convolutional decoding possible which in term made reliable transmission possible over wireless channels.

(Refer Slide Time: 41:04)



So, with that I think I would like to once again; and of course, the decoded information bits sequence, you can also see from the path that the information bit sequence, remember. The information bit sequence again from the path you will have 1 1 0 1 1 0 0, this is a maximum likelihood m l. So, these are the m l decoded information and this 1 1 0 0 you can verify that this is the decoded information bit sequence.

So, with that what I would like to do is I would like to stop this module here. And as I have already told you this module illustrated the concept the working of the procedure of Viterbi decoder through an explicit example by carrying in out a illustrating it trellis through a several stages. The procedure is likely complex and also can be confusing at time. So I urge you to again go over this module, if need be as clear any doubts. And also try working through an example yourself to understand it to best possible next time.

So I hope, I have been able to explain these intricacies of Viterbi decoder through this example. We will stop this module here and look at other aspects in the sub sequent modules.

Thank you very much.