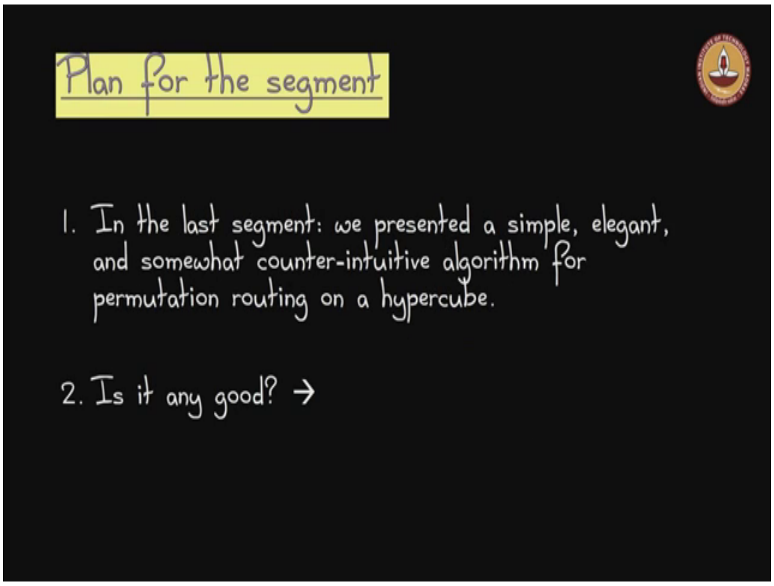
Probability and Computing

Prof. John Augustine
Department of Computer Science and Engineering
Indian Institute of Technology, Madras

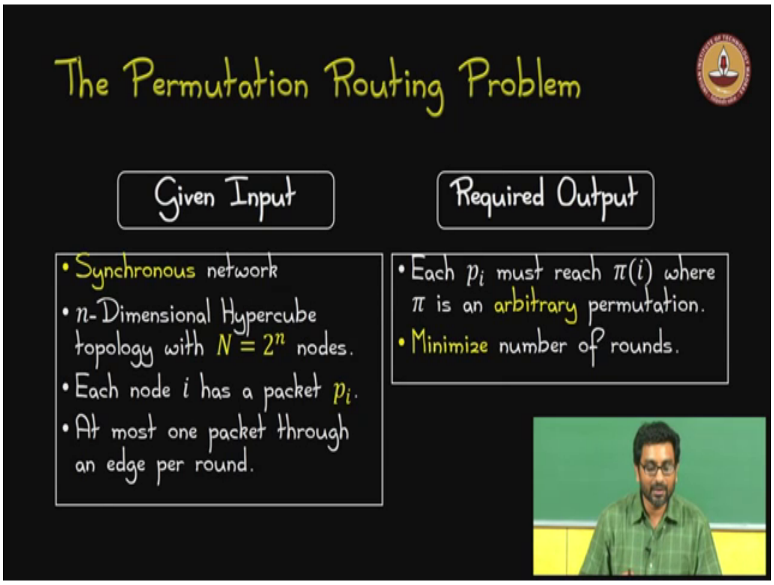
Module - 04
Application of Tail Bounds
Lecture - 23
Analysis of Valiant's Routing

(Refer Slide Time: 00:23)



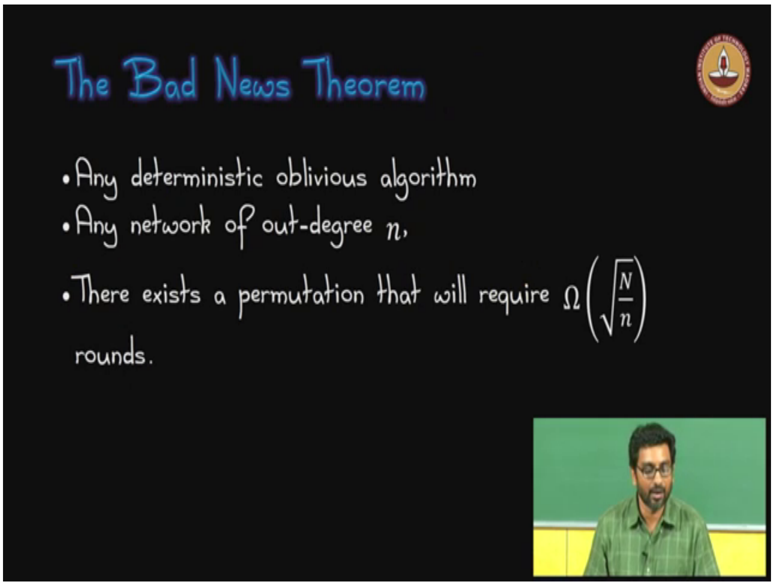
So, now we are in segment four of module four in which we are going to analyze valiant's routing which we saw in the in the previous segment. So, let us we need to ask ourselves if that algorithm is any good.

(Refer Slide Time: 00:27)



And let me quickly remind you of what the context is. So, we have a synchronous network on an n-dimensional hypercube each node i has a packet p i. And the packet p i must reach a destination node pi i, and pi is basically some arbitrary permutation and you must minimize the number of rounds, so that is the problem.

(Refer Slide Time: 00:56)

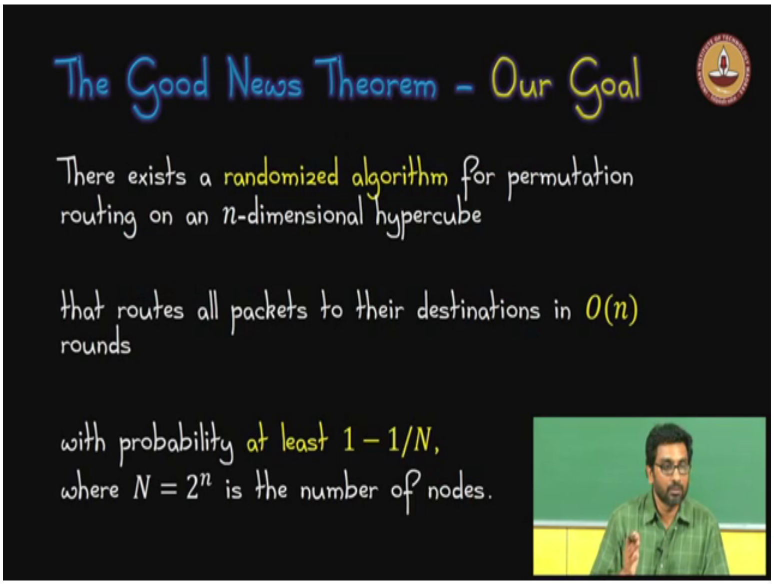


We have and recall that there was a bad news for deterministic algorithm basically there is a square root of n term that you cannot avoid.

(Refer Slide Time: 01:07)

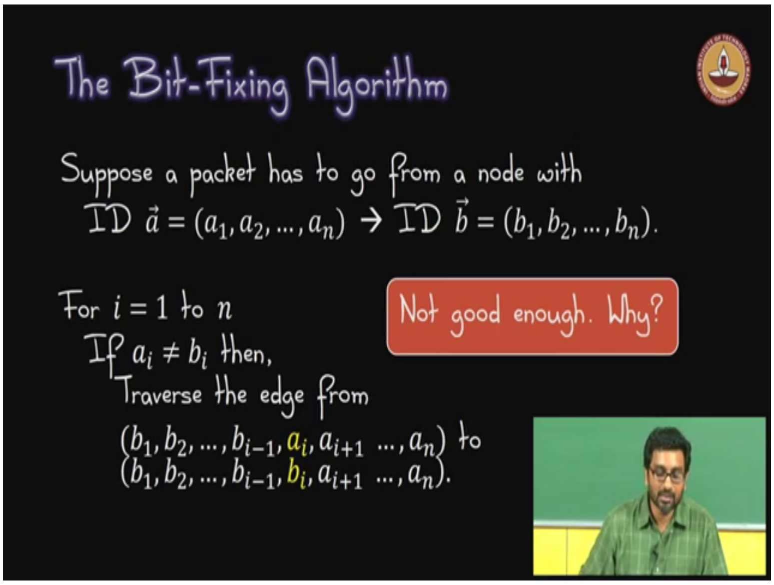
So, what we did was we randomized here we came up with a randomized algorithm.

(Refer Slide Time: 01:12)



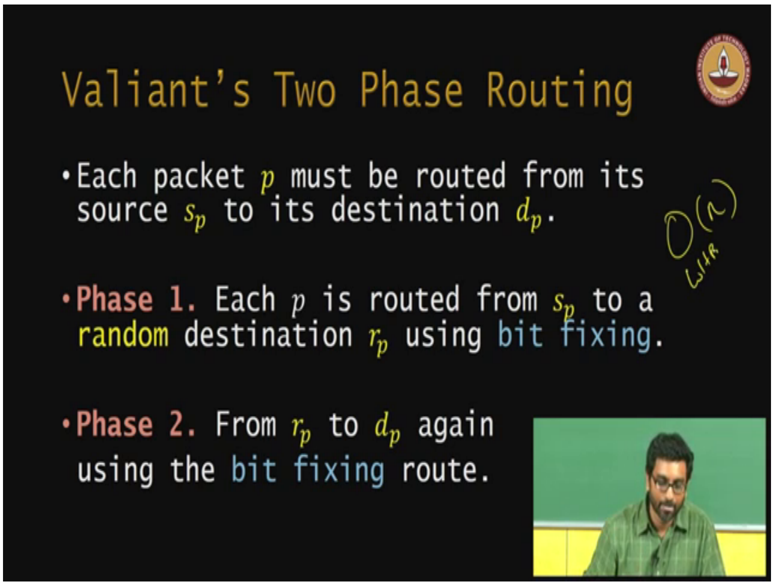
Basically the our claim which we have not proved, but we hope to prove in today's segment is that this randomized algorithm will complete permutation routing in O of n times and that will succeed with high probability ok.

(Refer Slide Time: 01:32)



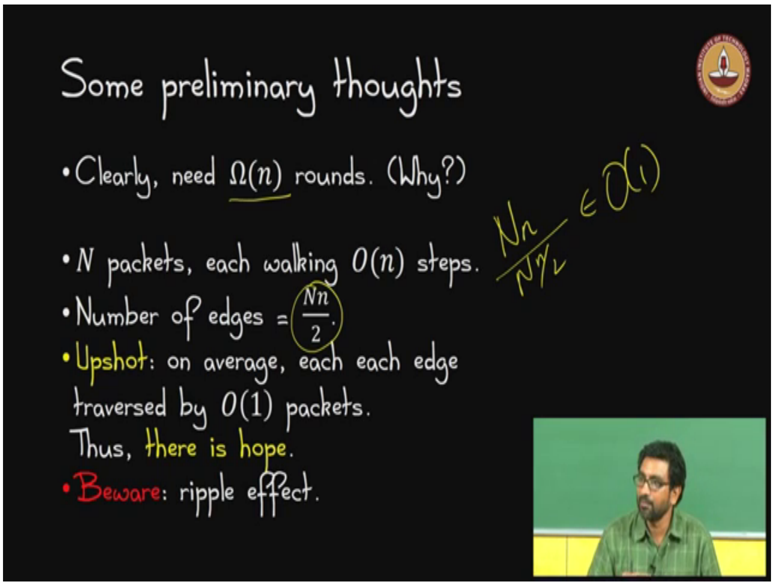
And what is this? So, well this randomized algorithm uses bit-fixing as a sort of a subroutine what is a bit-fixing basically just fix it is it is a way to go for a packet to go from one node to another node by just fixing bits one by one from left to right. But what is important is that is not a randomized algorithm, so deterministic algorithm, so it is not good enough.

(Refer Slide Time: 01:53)



Which means; that you need a slight modification this randomized algorithm where we go through two phases in which the packet goes from the starting node to a random destination using bit-fixing and then from that random destination to the final destination again using bit-fixing. So, there is these two phases that we need and this algorithm we claim has the nice property that it terminates in O of n rounds with high probability.

(Refer Slide Time: 02:26)



So, now we are ready for the analysis, some preliminary thoughts. So, one thing to start off with this clearly needs omega of n routs why is that yeah, but fixing is one specific thing, but you are only so the diameter of this network is n lowercase n which means and bit-fixing gives you the intuition as to why the diameter is n. And because the diameter is n you there can be always permutations where a packet has to traverse n edges and therefore, it will at least be omega of n rounds.

Student: (Refer Time: 03:02).

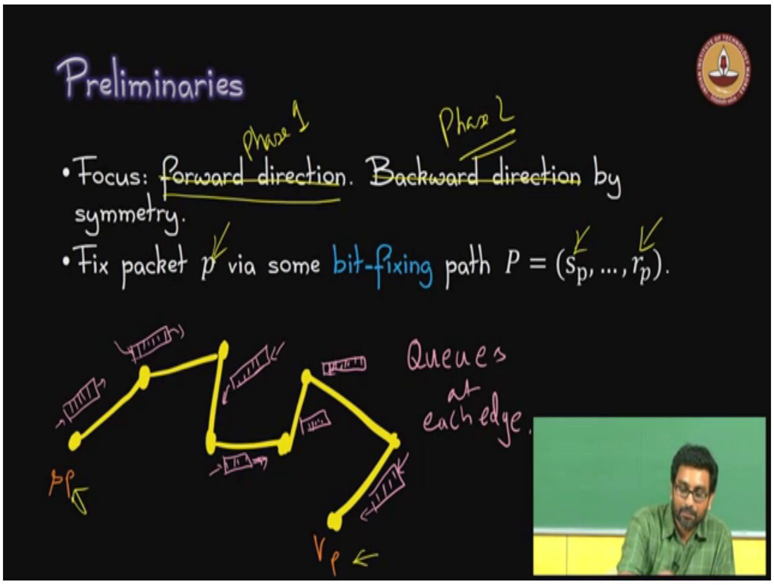
Yeah, that is yes that is the other thing. So, the random number is the destination, but even the source then the destination the adversary can actually decide say for example, the adversary decides that the 0 the packet at 00000 has to go to destination 11111 then all the n bits are to be flipped. So, let us let us look at it I mean one thing is there hope when we want to prove o of n we want to make sure that there is enough intuition to say that there is enough hope to even proceed forward ok.

So, let us try to establish enough hope that there is that we have. So, there are a capital N number of packets, each of them will have to walk for at most O of n steps ok. So, that itself means that there are capital N over lowercase n number of hopes that have to take place over all and how many edges are there well there are this many n edges. So, n is a lowercase n is a degree of each vertex, and there are capital N number of vertices. So, this many edges are there.

So, if you look at the number of hopes over the number of edges that itself is. So, this is sort of the good news if you will. In the sense that there are sufficient number of edges for you to get the packets across we do not know how, but at least if it were this where we do not even have sufficient number of edges will be in bad shape, we have sufficient number of edges. So, there is hope.

But the issue that we need to be aware of is that if is let us say one or two packets get stuck somewhere and they would caused a delay and therefore, a lot of other packets come and get stuck there. And then there is this ripple where a lot more and because of that delay a lot more packets get stuck and so on. This ripple effect is what we should be able to claim would not happen.

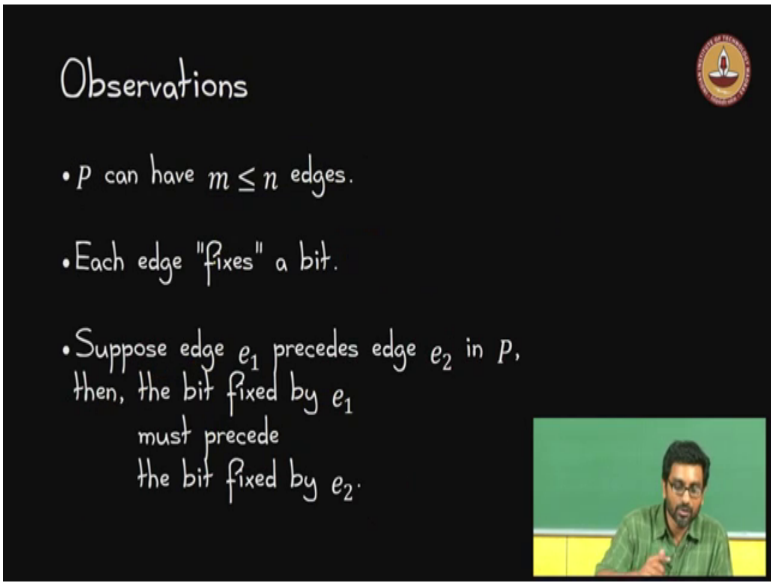
(Refer Slide Time: 05:13)



So, let us get into the analysis. So, some that start off some preliminaries something. So, one thing we are going to do is we are only going to focus on the forward direction phase one basically. In the back so not I would not say I guess I should probably not use the word forward direction phase one. And phase two, we are going to essentially hand wave in this ah segment, but it is an exercise for you to think about how whatever we discuss in class we will apply to phase two as well and complete the thought process needed to convince yourself that phase two also will work fine ok.

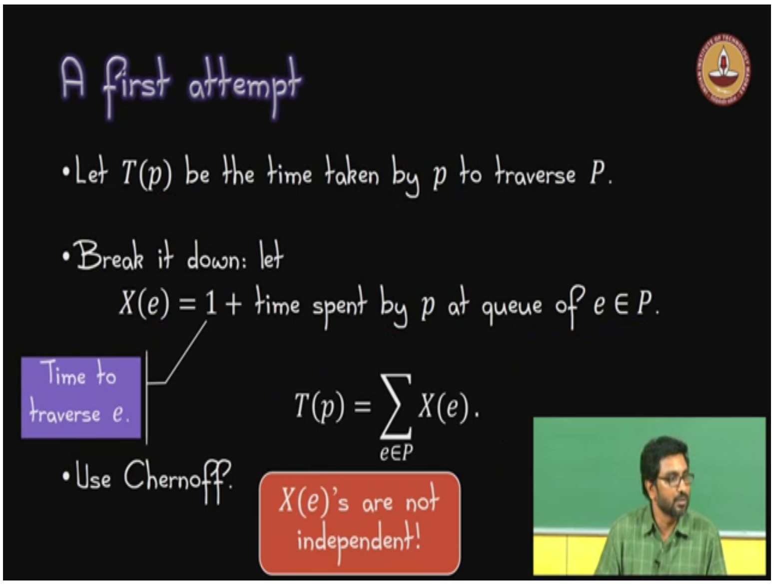
So, we are going to focus on phase one ok. And now in phase one let us focus on a particular packet p ok. This particular packet p is starting from s p and it is going to r p ok. And we have a picture for that. So, starting at s p and going to r p ok. And along the way in each edge that it traverses there is going to be a queue. So, it will have to if it is if it is if the queue is not empty it will have to get into the queue wait for it is turn to get out of the queue and then traverse the edge ok. So, this is the context.

(Refer Slide Time: 06:47)



This path this packet p uses we are going to denote that by uppercase P. And that can have at most n number of edges. Why, well because we are doing bit-fixing. And another important property of this thing if each of these edges are representing a bit that was fixed in sequence from left to right, so some of the bits might be skipped why because the current address and the destination already matched. But you never have it where there is a bit in the right side the that got fixed and then a bit for in the left side, you will always be fixing from the left to right all right.

(Refer Slide Time: 07:30)



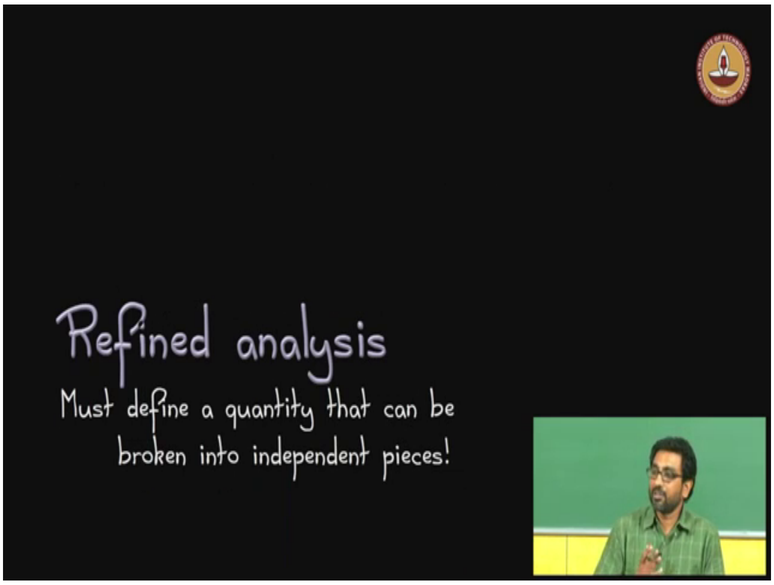
Let us make a first attempt and let us see where it takes us ok. We want to bounty p which is the time taken by p to traverse this packet p to traverse path P ok. So, now, how and whenever we want we have this large variable, we want to try and break it up into smaller pieces. So, how do we do that, we denote some we define something called X e remember capital P is made up of a lot of edges.

So, this e is one of those edges and X e is 1 plus the time spent by p packet p at the q in edge e ok. Why the one, one is the actual time it takes for that edge for the packet to go through that edge, but before it even went through the edge it at to spend some time in the queue that is taken care of by this part ok.

And now you can see that upper case T p is summation e over all the edges in that path in that path X e ok. So, we have a way to take this T p and break it up into smaller pieces and now hopefully we can use Chernoff bounds ok. What might be an issue with this, this is the problem here. So, these X e's there are well let us talk about a minor issue that possibly could be addressed first one is this is not a zero one variable ok, but that can be addressed.

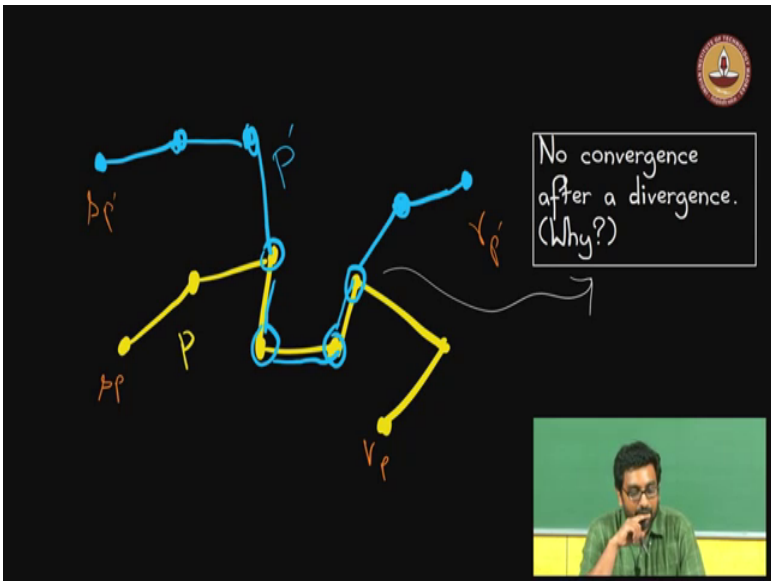
Because we can go back to the Chernoff bounds technique and massage it to work for this these types of small variables, but even if we did this the X e's are not independent and for Chernoff bound to work we need independence ok, so that is the problem ok, they are not independent. So, we can this will not work.

(Refer Slide Time: 09:24)



So, we have to refine the analysis somehow and in particular way we have to refine it. So, that we go after a certain quantity you want to try and define a quantity that can be broken into small independent pieces then we can use Chernoff ok. So, let us try to see how that can happen. So, let us get back to our path p ok.

(Refer Slide Time: 09:39)

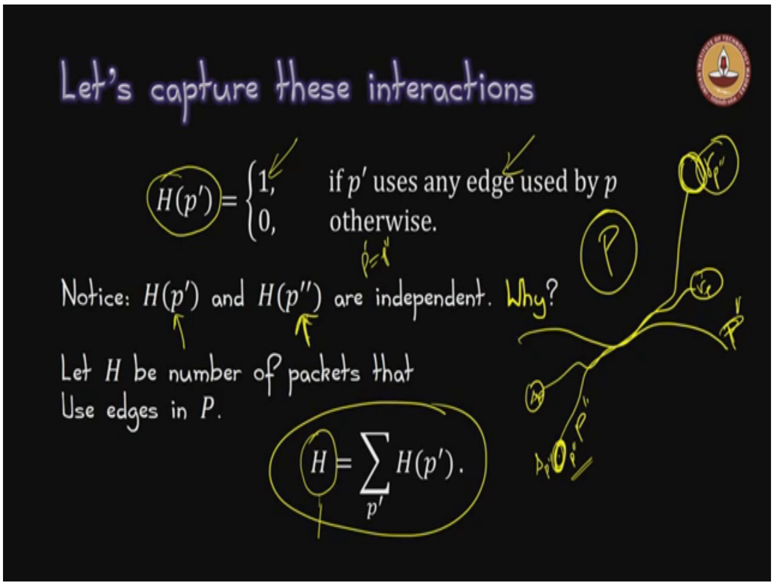


And let us try to see how that path P might interact with some other path P prime ok. So, now, you have a packet lower case p prime it is making it is way and that it is through a path uppercase P prime. Now, here is an interesting property here let us say well one possibility is that they never convert and they never say set of edges they used use are always destroyed that is great then; that means, there is no interaction. But if they were actually going to interact there would be a vertex at which they come together ok, they will converge at that point and they may follow each other for a little while, and then at some point they will diverge. Let us say they diverge will they have ever reconverge. So, we will never re converge because the moment they diverged that at that point they fixed different bits. And when they fix different bits their lef[t] left side of a.

Student: (Refer Time: 10:53).

Is always different their prefix is different which means that they are never going to be able to find a way to converge again ok. So, so this is a good property to keep in mind ok. It would not come in right now it will come in shortly, but easy to understand properly ok.

(Refer Slide Time: 11:09)



Let us now try to understand these interactions. Ok. So, now, what we are going after let me give you a sneak preview we are going after H which is the number of packets that interact with path p. So, this quantity we can try and break things down ok. How are we going to break this down, we are going to define the this was h parameterize by p prime and that is going be 1, if this other packet p prime uses any edge used by p basically at some point they converge, zero otherwise. And if now if you think about it H p prime and H p double prime when p prime is not equal to p double prime are independent.

Why are these two quantities independent h p prime is?

Student: (Refer Time: 12:00).

Student: (Refer Time: 12:01).

It is just another packet p double prime is another packet. So, if you have two packets H of one packet and H of the other packet are independent, basically if either of them interact with path p or not the permutation what permutation did we choose in phase one or was it even a permutation whatever it is the destinations.

Student: (Refer Time: 12:21).

It is random. So, for each packet the destination was chosen independently at random right. So, if you have a packet that is starting at something that packet in that node tossed n coins and that defined where the destination was. It had nothing to do with what some other random destination was chosen by another node ok. So, the paths chosen by p prime and p double prime are completely independent of each other. And therefore, with whether they interact with this path P that we are interested in or not is completely independent of for each other. This one is any edge used by P. And the reason is if it just uses a node then there is no delay caused by it ok.

So, now, once we have established this independence you know we can ensure that we basically establish that this capital H is the summation of these individual H p's, Hp primes rather yeah

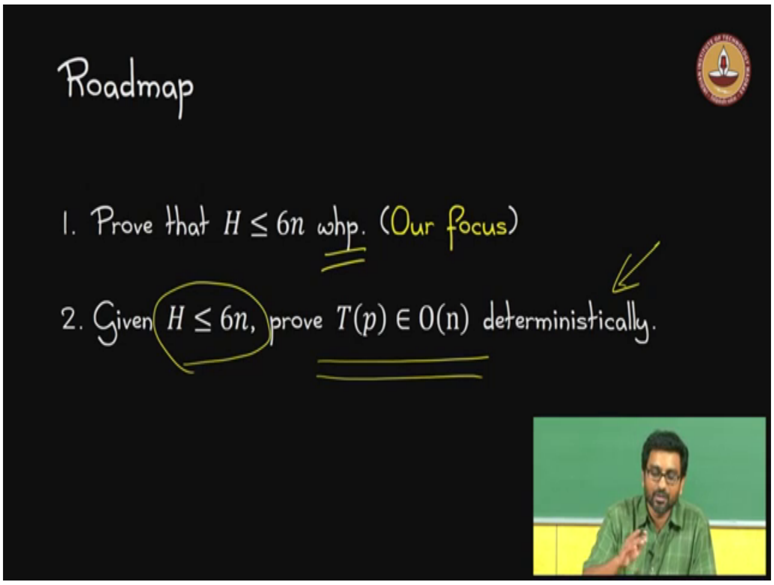
Student: (Refer Time: 13:25).

Ok, so let us what we care about is a path p ok. So, this is s p to r p this is our path p. And we are asking there is another this is the path P capital, P prime chosen by this guy this lowercase p prime. And then there is this other path p double prime chosen by p double prime. Well, in this case, I have drawn it like they are interacting with this path p that we care about, but whether they interact or not is independent of each other because what are these two paths.

So, let us take this path p double prime, it was this packet p double prime. So, it is basically source of packet p double prime all the way to the random destination of p double prime ok. And the source is fixed, but the random destination was chosen completely uniformly at random. So, now, the question is there is a certain probability with which this path will interact with this path p double prime will interact with p ok. But our claim is what is our claim; it is independent of whether p prime will interact with p or not ok.

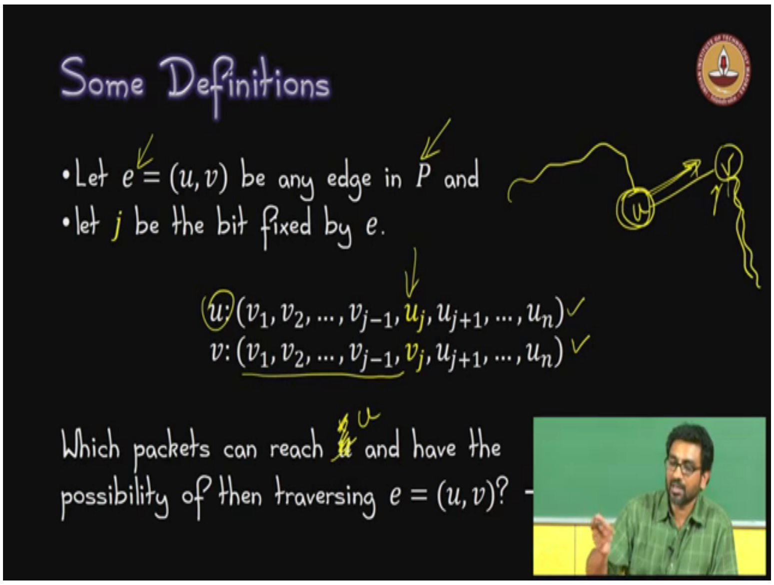
We are not claiming that their probabilities of interactions are equal, what are un unequal or anything like that, what we are claiming is this whether it will interact or not purely depends on the randomness over here ok. And as a result these two quantities H p prime and H p double prime are going to be independent of each other.

(Refer Slide Time: 15:33)



So, now this is our road map ok. We are going to prove that this H which is the number of packets that interact with this path p is at most 6 n and with high probability and then this is primarily what we are going to focus on. And then given that that is only a 6 n other packets interact with this path, we are going to prove that the overall time is going to be at most O of n. And this second statement is actually you can show a deterministic thing ok, so the first that is why we will spend more the first all right.

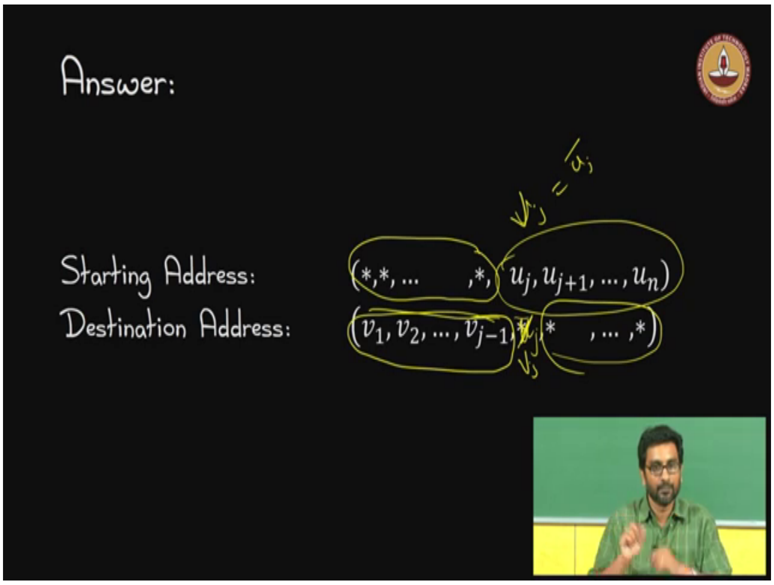
(Refer Slide Time: 16:11)



So, now so let us in order to do this let us get a few definitions down. So, now, remember we are focusing on a path capital P and e is some edge in that path and so that is denoted by this position here because that edge each edge is fixing some bit ok, it is fixing some bit and in this case it is fixing the jth bit ok.

So, I am going to have a ask a question which packets can reach v and have the possibility I am sorry can reach sorry it just gives which packets can reach u, and then have the possibility of traversing e. So, basically let us have a picture he is going from u to v; u has this address over here and v has this address over here ok. They differ in the jth bit. So, which packets can reach u go through v and move on. So, you need to be a little bit careful about the prefixes and suffixes.

(Refer Slide Time: 17:35)



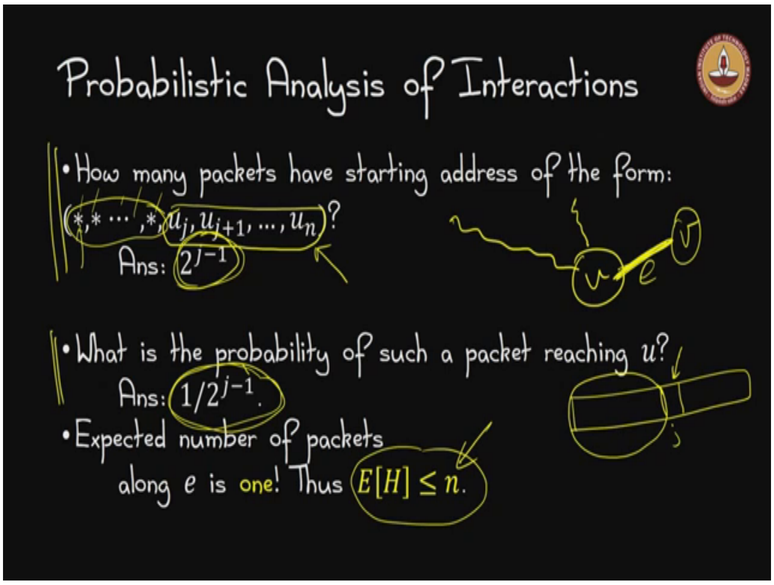
Well, let us be a bit careful here. So, which packets will reach u, it is not about their prefix, it is about their suffix if you think about it. Because the prefixes can somehow get bit fixed and changed and reach u, but if it had reached u and then here the jth bit was fixed ok, so then the prefix can be whatever they can be fixed, but all the suffixes u j on what should match that of vertex u do you see that. And it is destination here if you notice if it had reached u that means, it had fixed to make sure that these are all matching this one at this part.

And then after they traverses, so basically here ok, so here you can actually mention this as u j or something like that. So, basically here the destination should that particular bit should get fixed and then after that particular bit gets fixed then what it does after that we do not care. So, the suffix has to be there can be anything let me let yeah so basically yeah this is v j yeah you are right yeah. We are assuming in this case if it is actually going through an edge, we are assuming that u j is equal to v j, v j is equal to the complement of u j this is actually fixed.

Student: Two packets that are reaching the same vertex. They need not be reaching at the same bit right, they can be reaching at.

They can be reaching, but if they have to go through that edge they had to fix the exact same bit. There are to in this case they have to fix the jth bit if they were to go through that edge ok.

(Refer Slide Time: 19:39)



So, it let us let us proceed for now. So, here let us ask a few questions and try to answer them. How many packets, now we know the structure of the packets that go through that edge, how many packets have starting address of the form do not cares followed by u j, u j plus 1 and so on that is of course, 2 to the j minus 1.

What is the probability of such a packet reaching u, well each of these bits should have been randomly chosen, so that they fix their way to this particular node right. So, that is; that means, each of them out of the two choices that the correct choice should have been chosen so that is 1 over 2 raised to the j minus. Think of the number of packets along e. So, even if you think of the number of packets that reached that particular vertex u it is at most 1 on x ok. So, let me make maybe pause to make sure you get some time to think about it ok.

So, let us let us go through the questions one by one. How many packets have the starting address of the form shown here, and why do we care about it? We are talking about those packets that can reach this node u and that is basically 2 raise to the j minus 1, because there are j minus one do not cares that we have. And all the rest these are all fixed you if you look at the number of packets that can come to that particular node u, you cannot play with these values, but you can you can play with these values and there are j minus 1 bits to play with. So, there are 2 raised to the j minus 1 options for that ok, so that is the answer to the first question.

So, now, let us look at the second question what is the probability of such a packet reaching ok. So, basically what does such a packet look like, it will it will have an address that looks like this basically the first j minus 1 bits we do not care, it has some it looks like something. And then the rest of them are fixed right. What is the probability that such a packet will make it is way to u? Well, in the bit-fixing sequence, each one of them has to be fixed just right, so that the address matches you exactly which means in the very first bit the first star whatever the star was the random value associated with the destination should match the first bit in u that happens it probability half.

The second bit also should match with second bit of u again probability half and so on. Each bit has to be the random bit has to be chosen appropriately and that will happen with probability half and there are j minus 1 such choices. So, this probability is 1 over 2 raised to the j minus oh they have to this is stuff we are talking about reaching you going through particular edge and going through.

Student: (Refer Time: 22:50) jth position only how can (Refer Time: 22:53).

So, we are concerned about the number of packets that go through that particular edge e, this is our edge e that we are going to focus on this.

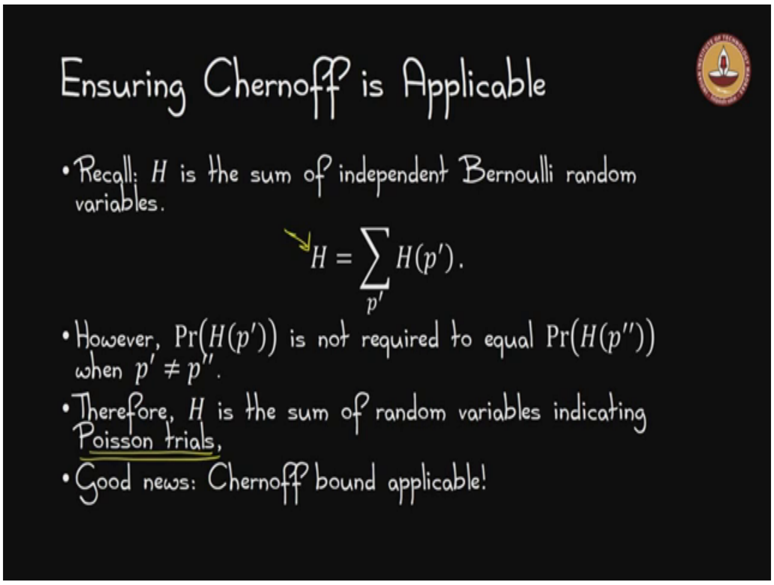
Student: (Refer Time: 23:05).

That is perfectly fine just this is perfectly fine because this one just happened to this one happened to be mostly matching u in terms of the bits that needed to be fixed right that is perfectly fine. But when it comes to u, what is needed is that if you look at this n bit string and this is let us say the jth bit, all of them should not change anymore only then the focus will be on the jth bit and that is important for us it.

Student: (Refer Time: 23:46).

Will let us take this offline if you need a little bit more ok. So, now, if you look at edge e that is one out of some n at most n number of edges in the path p. So, what is the expectation of the number of other packets that interact with that packet that that path p that is that will be at most n because you are adding that over all the edges ok.

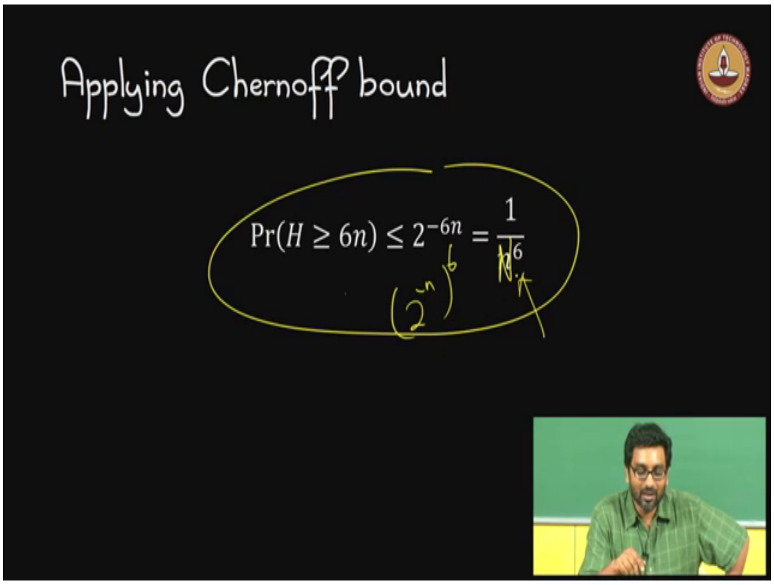
(Refer Slide Time: 24:15)



So, now we need we know we have this quantity H, we have we can we have established the expectation of H at least an upper bound on that and we need to ensure that we can apply Chernoff bounds. We know that the individual H p primes that add up to h. What are they? They are independent.

So, we also know that. And we know that they are not making any claims about their probabilities at this point. So, their probabilities each of the probabilities of H p prime equal to 1 or o or some quantities we do not they are different for each, but that only means that they these are this H is the sum of random variables indicating poisson trials; which is perfectly fine because our Chernoff bounds are applicable here ok.

(Refer Slide Time: 25:03)

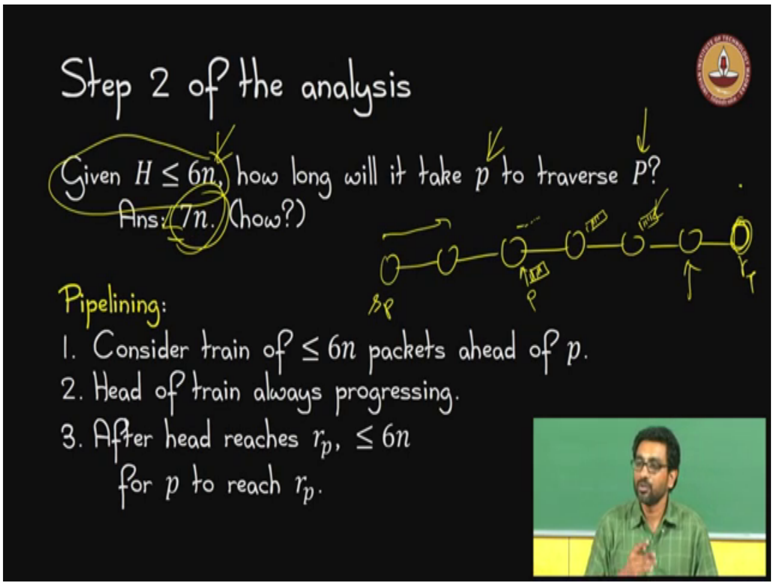


Which means that is it the slide has only one inequality; Probability that H greater than six n is at most two raised to the minus 6 and so this is the third inequality that I gave in the Chernoff bounds and that is see how low this is this is one over. So, and mu is at most n and that is so then it is at most 2 raised to the power minus 6. So, this is just 2 raised to the minus n.

Student: Capital.

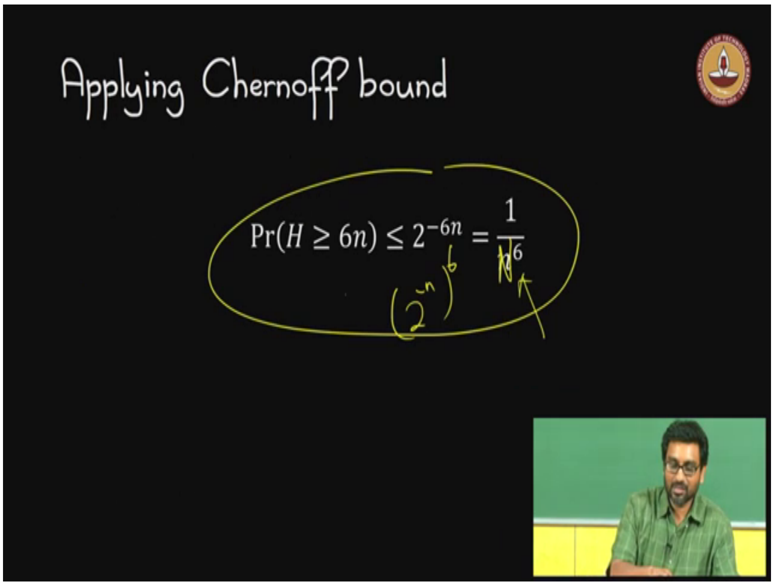
So, 2 power minus n is nothing but 1 over n the whole raised to the power 6 ok.

(Refer Slide Time: 25:42)



So, now let us see the second step in the analysis.

(Refer Slide Time: 25:47)



What we have done is, we have established that basically H is at most is at most 6 and with high probability it exceeds 6 n with very low probability. So, now, we are just going to assume that H is at most 6 n and how long will it take p to traverse this path capital P. And the answer is some 7 n and why because what is happening is a pipelining effect that is going on here ok. So, now, let us look at this path p ok. So, it is from s p to r p and the number of other packets that is use any of these edges is at most 6 n ok. Let us assume the worst case that whenever this p reaches a particular vertex, and it is in some q ok.

So, there are some packets that have gone past it, but if there is any contention it always gets the worst; I mean it is remember the you asked this question right how do our contentions result, p gets to be the last guy in such contentious. When a bunch of packets reach a particular vertex at the same time ok, there are already some packets in the queue, but when they get added to the queue at the end of the queue, p gets always unlucky it is the last guy.

But when you think about it there is a nice pipelining effect that goes on here. So, let me see how you can explain this. So, let us say p is in this location at this point. There are a bunch of packets waiting in the queue over here. There are a bunch of packets waiting in the queue over here, a bunch of packets waiting in the queue over here and so on ok.

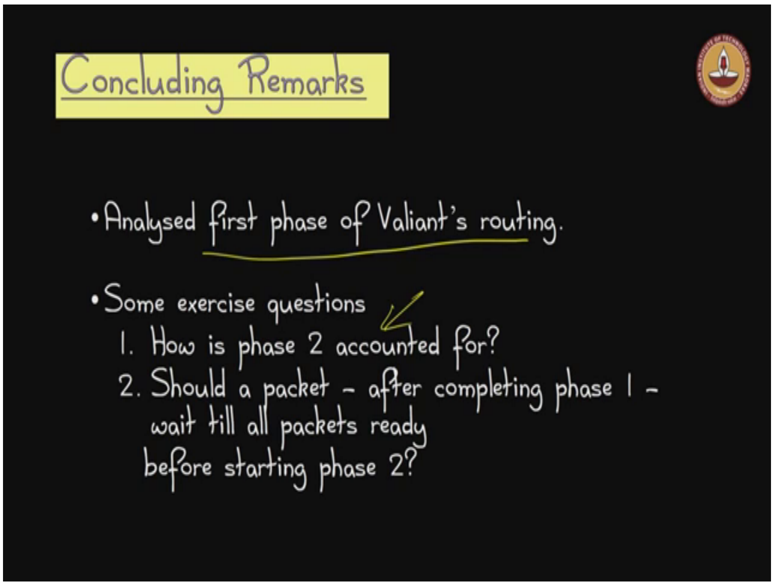
Consider the train and let us say there is nothing waiting over here this the this I call it a train where there is p and ahead of p there are all these packets in a series of vertices and then at some point there is an either empty thing or it is touching r p the destination ok.

What can you say about the head of the train in each time step the head of the train is this guy at the very beginning of the of this train. In each time step that packet is going to move one step forward or there are other I mean, I am hand waving here, but or the train itself could get a little longer because some packets joined and things like that. But if you look at it the head the position of the head will always keep moving forward ok.

So, if that is the case if the head started at sp and it always kept moving forward in n rounds it is the head would have reached r p ok. After our r p is reached by the head of the train in every time step what will happen one at least one packet from the head of the train is going to be knocked into the r p it is going to reach r p. How many such packets can get into r p at most six n and if that happens at that point in time p would be able to enter into r p.

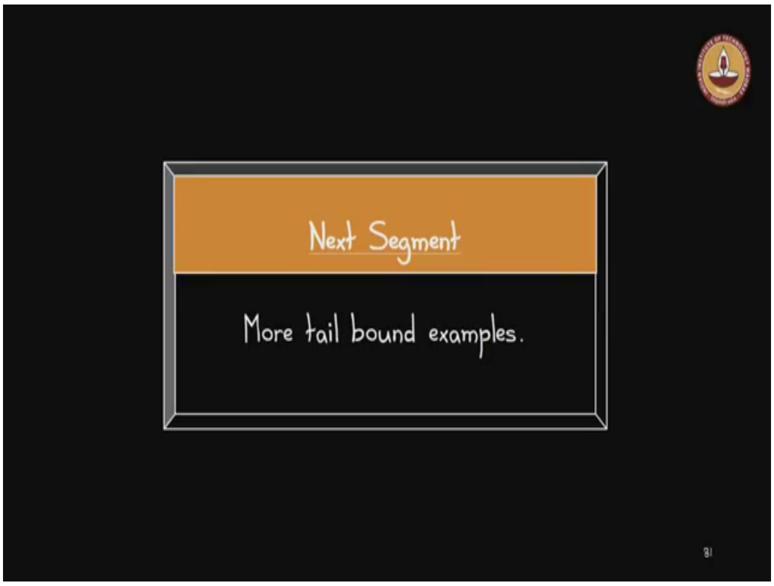
So, it takes n rounds at most for the head of the train to hit r p and then 6 n and rounds for the entire train to fall into r p is that argument somewhat clear ok. And that establishes that in at most 7 n and rounds your packet p is going to reach it is destination ok.

(Refer Slide Time: 29:33)



So, with that we will conclude what we have done is we have analyzed the first phase of valiant's routing. Some questions remain, so, for example, we have not talked about phase two. So, the other question that we have we discussed a little while ago is let us say a packet finishes it is face one, should it wait for all other packets to finish their face one. No, it can start it is face two and why is that again that is something you need to convince yourself ok.

(Refer Slide Time: 30:07)



So, I leave you with the those two questions and we will see some more examples in the next segment or so ok.