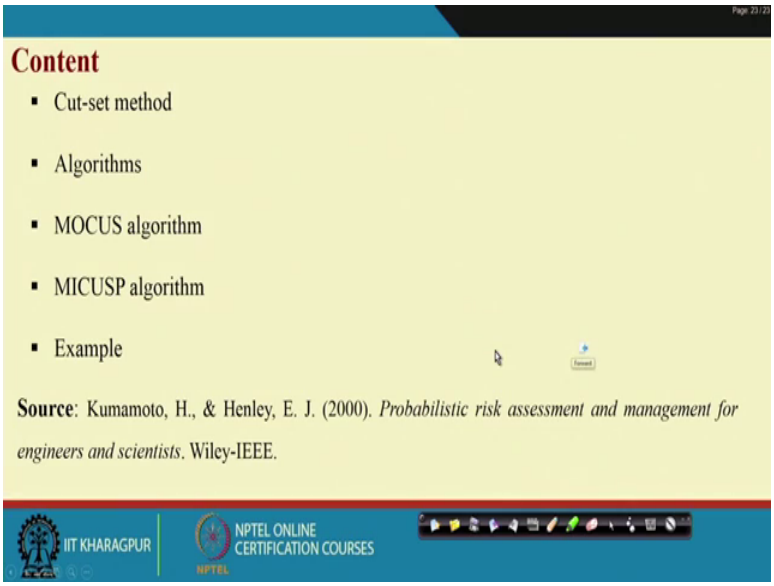
Industrial Safety Engineering
Prof. Jhareswar Maiti
Department of Industrial and Systems Engineering
Indian Institute of Technology Kharagpur

Lecture - 14
Fault Tree Analysis (FTA) - Cut- Set Method

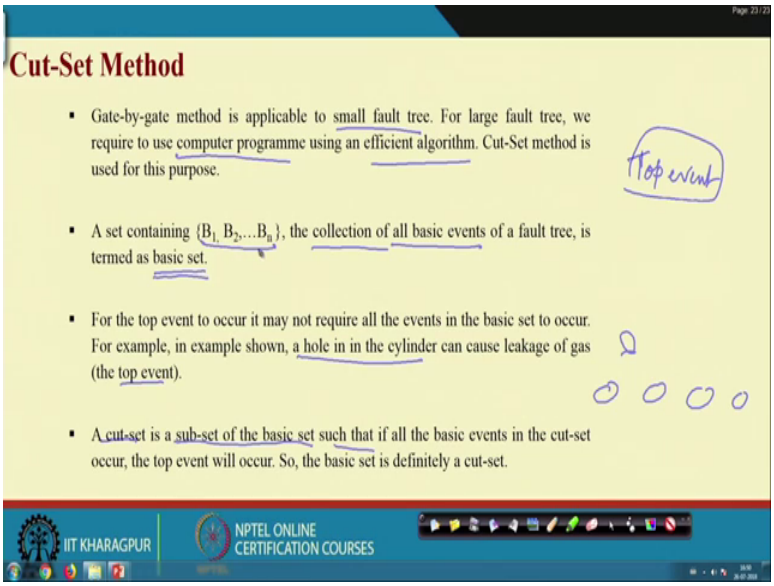
Welcome we will continue Fault Tree Analysis. Today we will discuss primarily Cut-Set Method.

(Refer Slide Time: 00:23)



So, the contents of today's presentation one the conceptual issues related to cut set then the algorithms MOCUS and MICUSP algorithm. We will be discussing primarily the MOCUS algorithm, and there will be example on cut set methods. And the source is the same book that is Kumamoto and Henley, probabilistic risk assessment and management for engineer and scientist it is Wiley-IEEE.

(Refer Slide Time: 00:56)



So, you have already seen the gate by gate method in gate by gate method, the top event with different gates finally at the bottom events. The bottom event probabilities is known, and every gates are quantified and finally, top event probability is quantified. So, that method is good for small fault tree. For large fault tree, you required to use computer programming and you require an efficient algorithm for that purpose.

Cut-set method is one such algorithm that will help in quantifying the top event probability through computer programming. So, there are certain terms, and certain issues related to this cut set method. For example, first one is the set. So, all of you know the set is collection of items, collection of individuals or collection of entities. In fault tree it is collection of basic events, all basic events.

So, a set continuing all the basic events in a fault tree is termed as basic set. It is set because it talks about the event related to all the component level failures. So now, if as you have already seen from the gate by gate method, if you know the probability of the basic set, means the component probability, you will be able to find out the top probability top event probability. So, and also you will see that for top event to occur, it is not that all the bottom event should occur. May be a combination of bottom events will ultimately lead to top event.

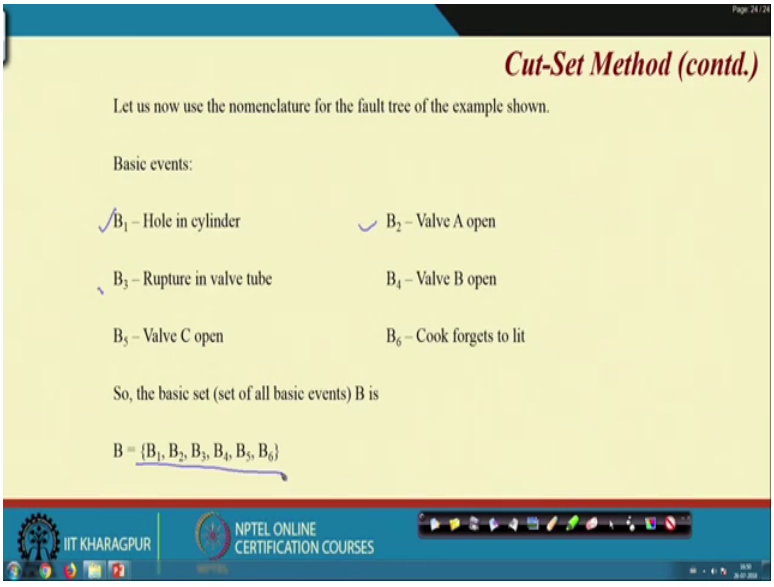
So, that is why for the top event to occur, it may not require all the events in the basic set to occur. Now we are coming in terms of talking in terms of set. So, basic set means

what? All the basic events so, for top event to occur all the basic event like B 1 to B n, that does not required to occur may be one or more in combination, ultimately lead to the top event occurrence.

So, for example, a hole in the cylinder can cause leakage of gas that is the top event. Means, only one basic event and whole in the cylinder particularly the kitchen, gas cylinder a hole can cause leakage of gas. So, one basic event may be of such capacity that it can lead to the top event occur or a combination of basic events finally, lead to the top event occur.

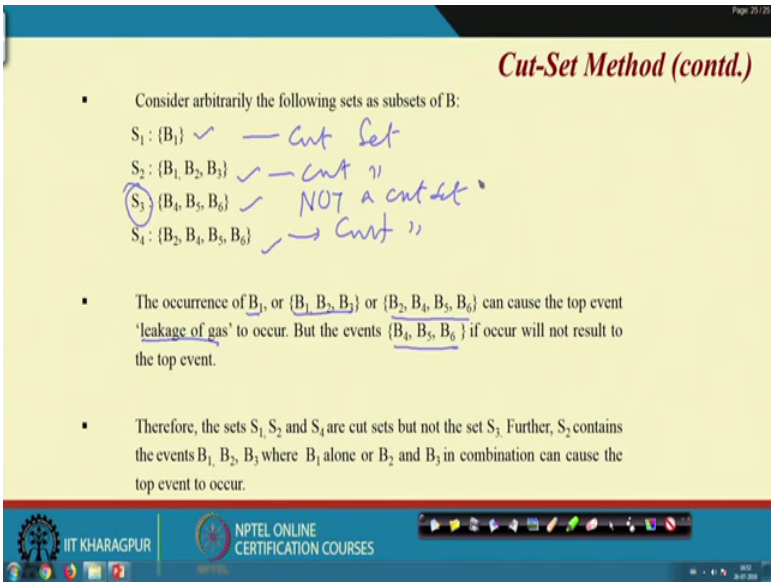
So, that what is cut-set? A cut-set is a subset of the basic set such that if all the basic event in the cut-set occur then the top event will occur. So that mean what we say? That basic set is the ultimate that is a superset. So, there may be subset that can be derived from the basic set, and what is the property that if all the events in that subset occur, then top event will occur, then that subset is cut-set.

(Refer Slide Time: 04:50)



Now, let us consider the example. All of you have seen this example in earlier class, that we have 6 basic events B 1, B 2, B 3, B 4, B 5 and B 6. And then what will be the basic set? Basic set content all the basic events.

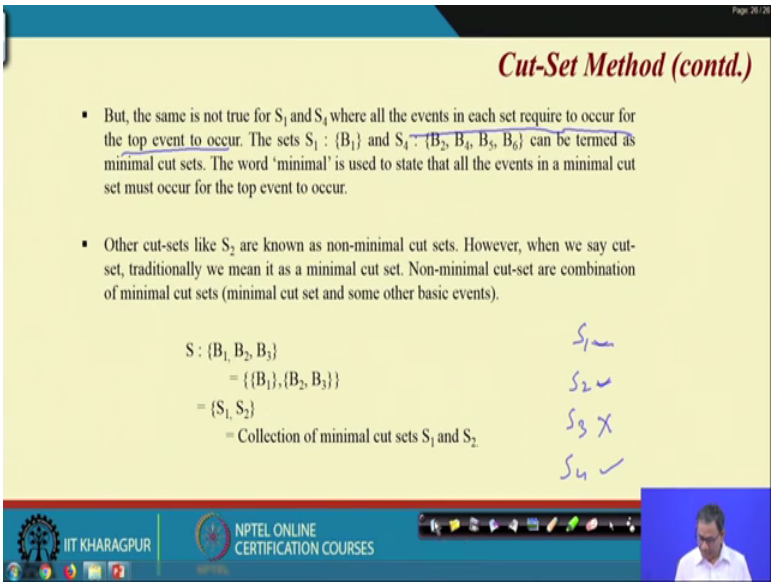
(Refer Slide Time: 0:14)



Now, let us arbitrarily create some of the subset, only B 1 B 1, B 2, B 3, B 4, B 5, B 6. B 6, B 2, B 4, B 5, B 6. So, you have 6 basic events so, basic set content 6 elements. Now arbitrarily we have created this many subsets now for the sake of understanding, we will examine one at a time. And already the cuts the total fault tree utilising the 6 basic events I have given to you, just follow that fault tree. And you will see that from that fault tree, the occurrence of B 1 alone or B 1, B 2, B 3 in combination, or B 2, B 4, B 5, B 6 in combination can cause the top event that is leakage of gas to occur, ok.

So, you have seen this one. But the S 3 which is basically B 4, B 5, B 6, if the event B 4, B 5, B 6 occur the top event will not occur. So, as a result S 1 is cut set, S 2 is also another cut-set. S 4 is also a cut-set, but this one S 3 is not a cut-set. I hope you understand, ok.

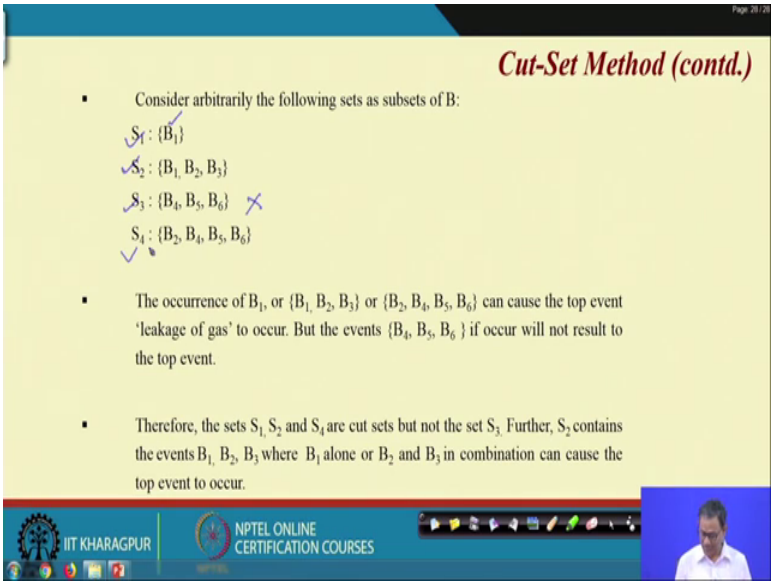
(Refer Slide Time: 06:59)



So, now what we have found out? We found out that the from the arbitrary sub selection of subset like S 1, S 2, S 3 and S 4 this is cut set, cut set, not cut set, this is cut set,. So, what happen in S 1 and S 4? All the events in the set required to occur for the top event to occur let me go back again.

For example, for example, let me go back again.

(Refer Slide Time: 07:41)

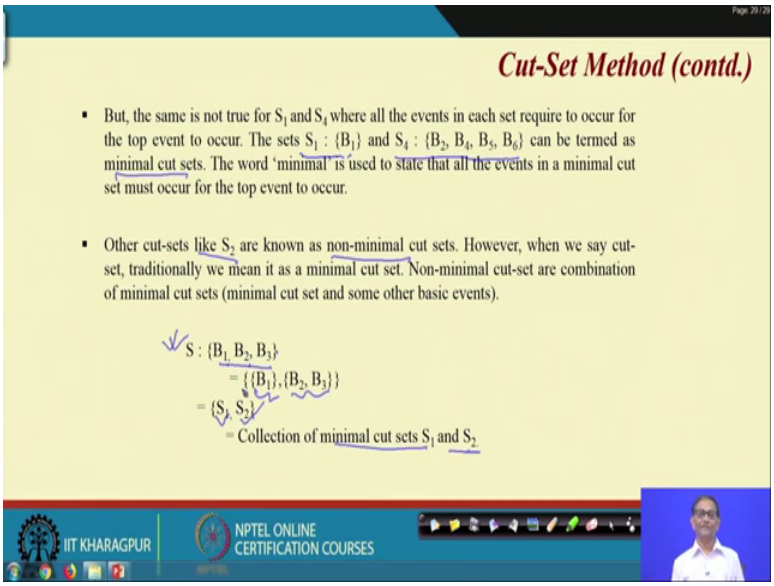


For example, S 1, so, B 1 must occur. S 3 B 4 B 5 B 6, if that occur this will this is not a cut set, S 4 and S 2. So, you will find out that ultimately, I told you that here this must

occur and here in S 2 you have to see that whether all the 3 required to occur, or may be a combination will another combination will help similarly S 4. So, that is what I want to discuss now.

What I want to tell you, that what we have written here.

(Refer Slide Time: 08:27)

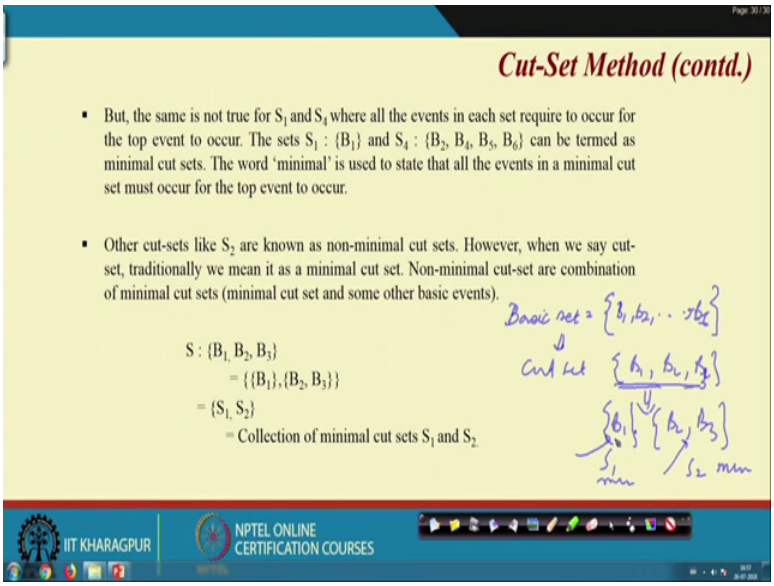


That that the set S 1 which basically contain B 1, and your S 4 B 2, B 4, B 5, B 6 can be termed as minimal cut sets. Why? Because all the elements in minimal cut set must occur that mean, then only the top event occur. But if you take the cut set S 2, that is non minimal, because this S 2 that B 1, B 2, B, B 3 and here what happen? B 1 and B 2 B 3 then, this S 1 and S 2 both are equally able to help occurring the top event.

So, as a result what happened? 2 minimal cut set S 1 and S 2 and ultimately that S 1 if occur and S 2 occur, they independently can make the top event occur. Now if you create a set like this B 1, B 2, B 3 this is not minimal, because all the events in this set does not required to occur for the top event to occur.

So, then what happen basically? We have given you 3 things; one is basic set.

(Refer Slide Time: 09:52)

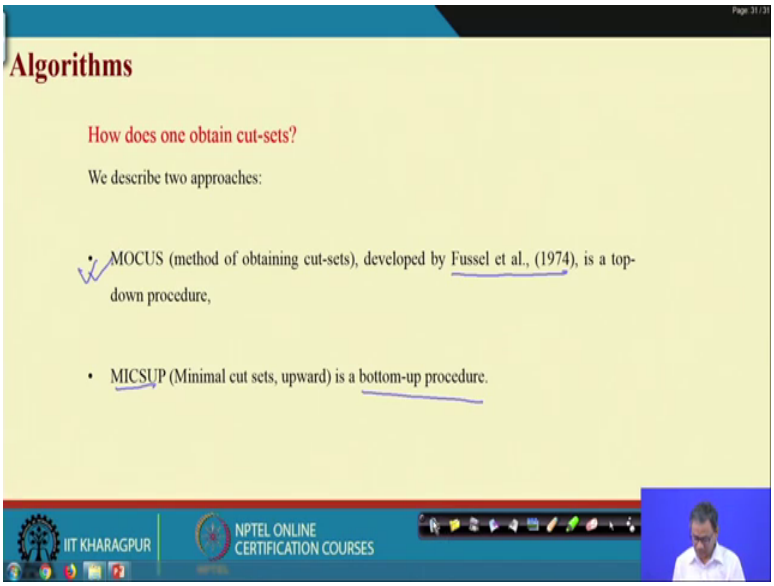


For example, B 1 B 2 to B 6, then we say cut set. For example, B 1, B 2, B 3 this is a cut set. Why it is cut set? Because this 3 basic event if occur, the top event will occur. It is not basic set because basic set contents all the bottom level events.

But this one can be written like this B 1 and B 2 B 3. Now B 1 alone is sufficient to cause the top event to occur so, that is why this is a cut set. Again B 2 B 3 sufficient to cause the top event to occur then this is a another cut set. The property of here that if B 1 for example, a 2 both B 2 and B 3 should occur then the top event occur, that is why this is minimal cut set.

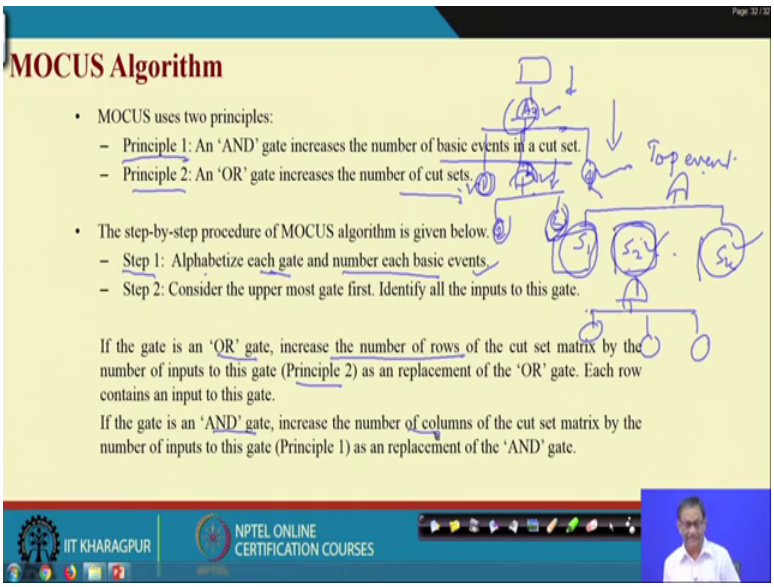
For example, S 1 is minimal although it has one only one element, but these cut set B 1, B 2, B 3, it is not minimal. Because it is combination of 2 minimal cut set. So, we have basic set, we have cut set, we have minimal cut set. So, we want to find out the minimal cut set. Usually, when we say cut set method we try to find out the minimal cut sets.

(Refer Slide Time: 11:38)



So, how do you generate cut set? We will use MOCUS algorithm developed by Fussel et al 1974. It is a top down approach or mix up; that is minimal cut set upward it is a bottom up approach. Both the approach are more or less similar. So, but one is from top to bottom another from bottom to top so, we will discuss this approach.

(Refer Slide Time: 12:13)



Please listen very carefully that how mix MOCUS algorithm works. So, it is very important algorithm for fault tree analysis. There are 2 principles. Principal 1 and principal 2, it is basically related to AND and OR gate. So, when you develop a fault

tree, you might be using other gates, not necessarily only AND and OR gate, you may be using other gate like Exclusive OR gate and priority AND gate some other gates. So, it is recommended that you convert other gates into equivalent AND or OR gate or combination of AND and OR gate. And then make the make the fault tree having only AND and OR gate that is the first thing.

And second is that you use 2 principal to find out the cut set one is principal 1. It relates to AND gates, if there is any AND gate, it increases the number of basic event in a cut set. So, let me write down like this suppose top event so, there are let there are many cut set write S 1, S 2 suppose S k, so many cut sets you know. What is cut set? If the event in this set occur, top event will occur, if event in this set occur top event will occur, if even in this S k set over top event will occur.

Then we can say it is OR gate with all the cut sets, or you can write like this. So now what we say? If there minimal cut set, then all the basic events here must occur then only this will occur so that mean there is an AND gate with the basic event related to this cut set similarly related to this cut set similarly related to this cut set, ok.

The principal is in the fault tree when you are coming down from the top to bottom, whenever you encounter an AND gate, it simply indicates that that increases the number of basic event it cut set it will not increase the number of cut sets. So, given a fault tree like this suppose this is my AND gate, then here is one basic event, here is one OR gate, here is another 2 basic events, and here is one basic event.

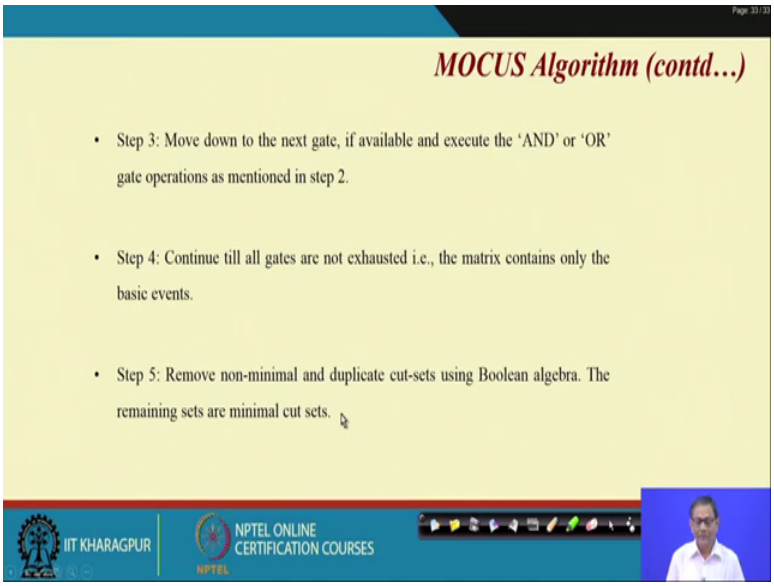
So, coming down you found out an AND gate, what does it mean? This gate will basically we will increase the number of basic event in the sets. So, this start with the set here, this one, this gate and this basic event will be the element and it will be only one set. It will not increase the number of cut set. But when you get an OR gate, it increases the number of cut sets. Because OR gates is more vulnerable. So, this principal you have to apply and we will see how it will work.

Then how do you proceed? Step one alphabetized each gate and number each basic event. So, that mean, if this is the fault tree gate will be A B. And number means like 1 2 3 4. That is why alphabetize each gate number of and number each basic events. Consider the upper most gate first identify all the inputs to this. First you consider this, see the inputs and accordingly use the applicable principal whether AND and OR, and

then either you increase the number of basic events in a cut set or number of cut sets.

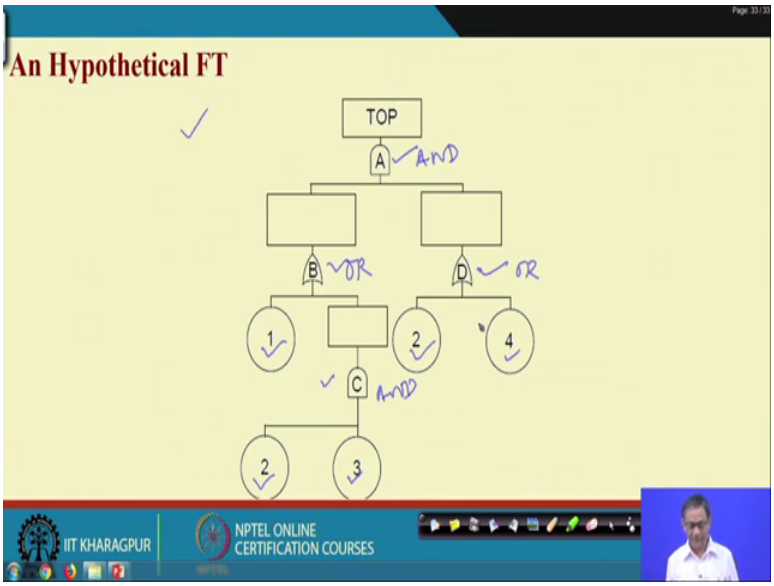
So, if the gate is OR gate increase the number of rows, and a that is principal 2 and if the gate is AND gate increase the number of columns. So, let me tell you what is this; I think next slide we will see what is this.

(Refer Slide Time: 16:56)



And then what happened? You move down to the next gate. And in this process go down to the to the bottom, bottom level and then you will having lot of cut sets. And then all non-minimal and duplicates cut sets will be will be removed using Boolean algebra, then the remaining cut sets will be the minimal cut sets.

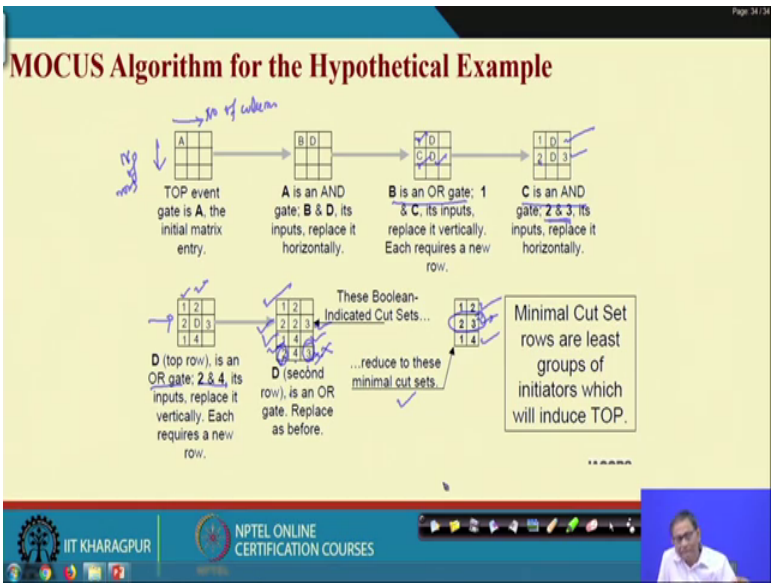
(Refer Slide Time: 17:23)



Suppose this is our hypothetical fault tree. Top event alphabetised gate A, gate B, gate C gate D. This is my AND gate, this is another AND gate, then this is OR gate, this is OR gate. Now number the basic events 1, 2, 3, again 2, 4; that means, this events coming in the 2 places of the fault tree.

So, we want to use the MOCUS algorithm now. How we will do?

(Refer Slide Time: 18:11)



You see, first gate is what? You create a matrix so, in the matrix, the number of columns and number of rows. So, number of rows number of columns so, rows transfer cuts

number of every row talks about a cut set. And every column talks about the element in that corresponding cut set means corresponding row.

So, we have already seen that the diagram we have already seen diagram is top event then A. So, start with this so, we are doing this, start with A. Now what is A? A this top event gate is A the initial entry into the matrix. What is A? A is AND gate; if it is AND gate what it will do? It will not increase the cut sets, increase the number of cut sets; that means, there will be, it will be in the same row there will be no additional go for these. And it increases the elements in the cut sets so, first cut set.

So, the input to A is what? Input to A is B and D, input A is B and D. So, that mean what will do then? A is converted to B and D in the first row, number of elements is increased. The reason is A is AND gate. Now see you have 2 gates here, B and D. Suppose, we consider B first. What is B gate? B is OR gate. So, if it OR gate, it will increase the number of cut set.

So, how many what will be the number increase; depends on the number of inputs to this gates. If the gate has 2 inputs, then the first that there will be 2 cut sets. If it is 3 inputs, 3 cut sets like this. Now B has 2 inputs. What are those 2 inputs? 1 and C. So, that is why B is replaced by 1 and C. D remain as it is because D is common to both 1 and C.

So, what happen? You see in the AND gate case the element in the row increases, in the OR gate case number of rows increases. Number of rows is synonymous to number of cut sets, number of column synonymous to number of elements in a cut set. Then there are 2 things C and D gate, now you can choose C or D anyone, let us go by alphabets order so, C is chosen.

What is C? C is an AND gate. If C is an AND gate, it will not increase the number of cut set rather, it will increase the element in the cut set. So, the input to C is 2 and 3, and as a result you see 1 D as it is, now C is replaced by 2 D take each place 3 is added here one more, one more column is added. So, that is why 1 D, 2 D, 3 this is basically the cut set, 2 cut sets.

So, how long you will continue? You will continue so long the matrix will be with only numbers, there will be no alphabets because alphabets, talks about the gates. Now see how many alphabets are there? Only one that is D, but in 2 places so, we will write what

is the input to D input to D is a 2 and 4. So, and what is D? D is OR gate, if D is OR gate it will increase the cut set.

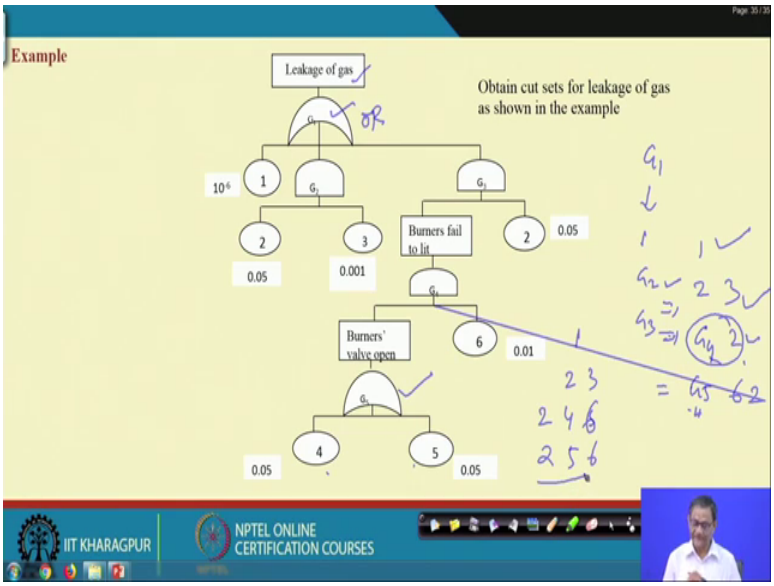
So, what happened? 1 as it is, then D is replaced by 2, 2 ok. The second row we kept as it is, but as D being the OR gate it increases the number of rows. So, 4 2 already we have considered we have to consider 4. So, that is why 1 and 4; 1 is common, 2 and 4 are input to this gate. This is first time we are doing. But there is that D is not completed here. So, there another D is there. In the same manner, you have increase the number of rows. So, as a result what happen? You got one cut set 1 and 2, another one 2 to 3, another one 1 4 another one 2 4, 3.

Now, you use Boolean algebra to reduce this to minimal cut set. Like 1 2 it is minimal, because both required to occur. 2, 2, 3 this is nothing but 2, 3 because the same basic event coming 2 times, it is Boolean algebra concept is 1. So, the second one is reduce to 2 and 3. Then 1 4 as it is. Now what will happen to 2 and 4? 2, 4, 3 this is not a minimal cut set. The reason we have already seen, that if 2 and 3 occurs top event will occur, now here 2, 4 and 3 is there. So, 2 and 3 already there so, that mean that is already considered. So, this is not minimal and it is not required so, we have removed this one.

So, as a result the number of minimal cut set is 3 reduced to these minimal cut set. This is what is the algorithm MOCUS algorithm for minimal cut set ok I hope you have understood. Let me repeat again we are using 2 principal's principal one, AND gate will increase the number of elements in a cut set. Principal 2 OR gate increases the number of cut set.

What will be the increase? In both the cases increasing the number of inputs to that particular gate; if it is AND gate, then the number of element in the cut set will be the number of inputs to that AND gate. If it is OR gate, number of cut set will increase and that increases number of input to the OR gate. And using this principal and then to make it programmable, what we have done? We are basically saying that you alphabetize the gates and number the basic events, when then the follow this algorithm. I hope you understood.

(Refer Slide Time: 25:13)

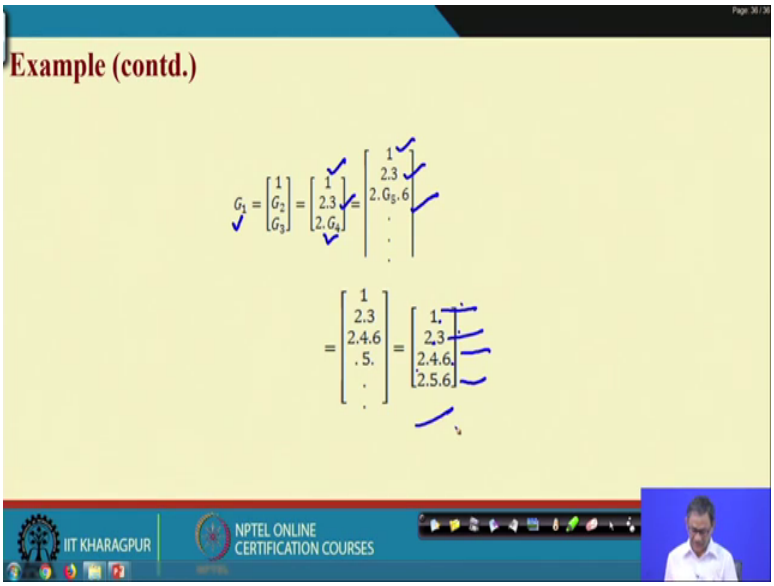


See the situation here leakage of gas OR gate, how many inputs? 1, 2, 3 so, if you want to create so, first you start with G 1, then it is OR gate, it will increase the number of rows 1, G 2, G 3. So, these are all cut sets.

Now, you talk about G 2. So, G 2 is what AND gate so, input how many inputs 2. So, what will be the next is then? 1, G 2 will be replaced by 2 and 3, and G 3 will be as it is. But G 3 is again you see it is basically G 4 and 2. So, you write this one, and it is an AND gate. So, G 4 and 2.

Now, what is G 4? G 4 is another AND gate, and which at G 5 and 6. So, that mean G 4 will be replaced by, then this will be there this is, then G 2, G 4 this portion is replaced by G 5 and 6. G 5 is OR gate having 4 and 5. So, that mean resultant things will be 1, 2, 3 then 4 4 6 [FL] 2 is there [FL] ok. G 4, G 4 2 is there. So, then 2, 4, 6 and then 2, 5, 6 using MOCUS algorithm, you are getting this, it is there.

(Refer Slide Time: 27:00)

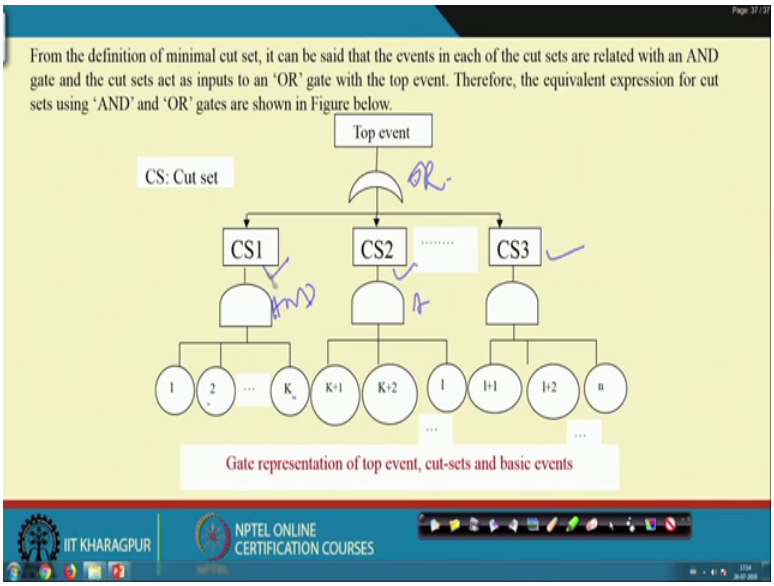


You see it is there, G 1 is OR gate, 3 inputs, 3 cut sets. Then 1 as it is, it is basic event nothing to do G 2 is AND gate with 2 basic events fine, 2 3 G 3 having basic event 2 AND gate 4 so, like this.

Now, 1 as it is these are G 9, G 4 is AND gate with G 5 and 6. So, 2 G 5 6 in this manner, break and ultimately, how many cut sets you are getting 1, 2, 3 and 4. Now there is no redundant basic events so, 1, 2, 3 there is there are unique 2, 4, 6 units 2, 5, 6 units.

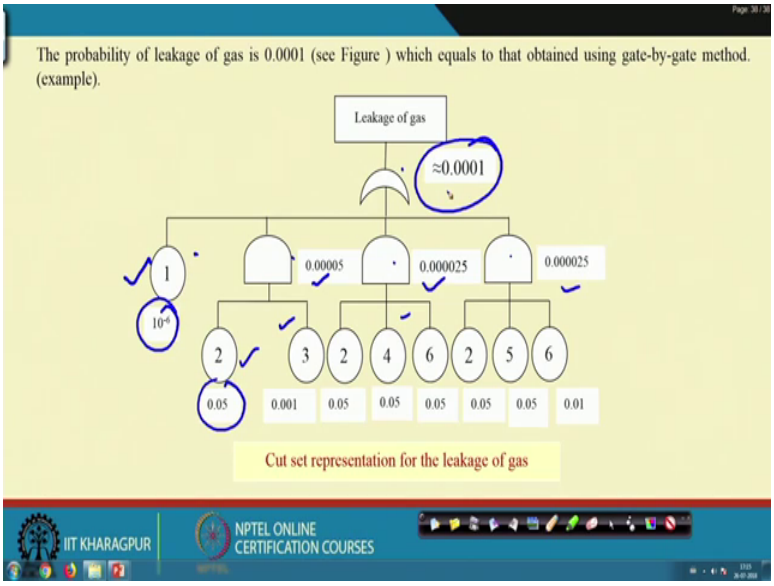
Now, what you have to see? You have to see that whether they are basically minimal or not. See one no (Refer Time: 28:00) 2, 3 is not sufficient to replace this 2, 4. 6. Not sufficient to replace this also. So, this is a minimal cut set so, this is what is the concept of cut set MOCUS algorithm.

(Refer Slide Time: 28:14)



So, once you have the cut sets ready, then the logical linking is top event with OR gate. And this is cut set 1, cut set 2, cut set 3. And cut set means minimal cut set. So, all the basic events should occur, that is why AND gate, AND gate like this; where we once you know the number of cut sets, and the element of the cut sets, you will be able to find the probability of this cut set, this cut set, this cut set. And then find out the probability of the top event using the OR, AND gate logic, ok so, this is what is our cut set method.

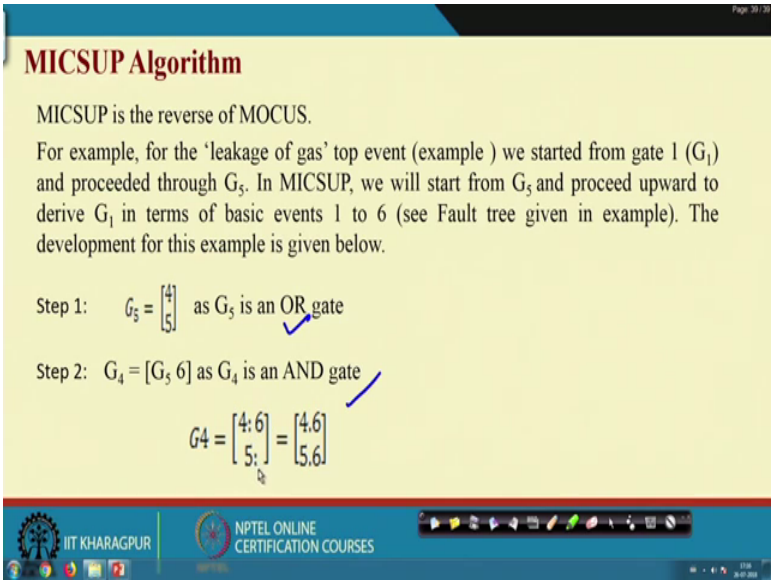
(Refer Slide Time: 28:59)



And for the example, we have already found out 1 is cut set, then 2 3, then 2 4 6, 2 5 6

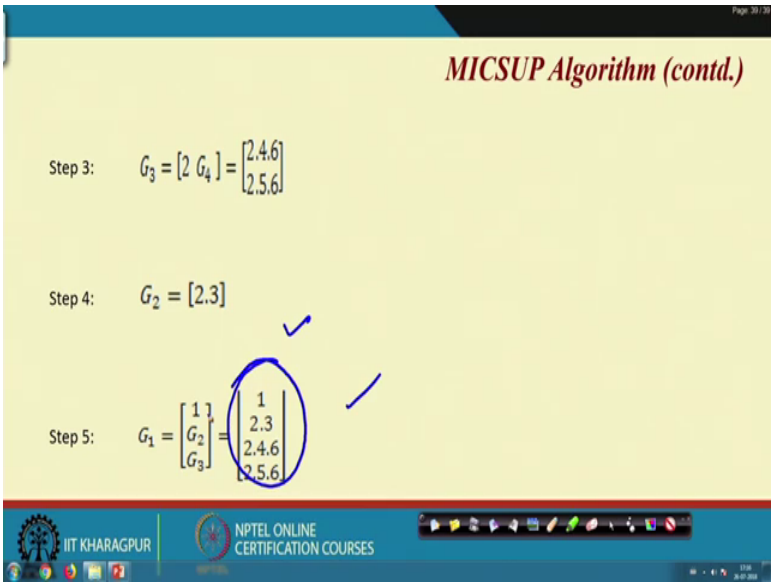
this probability known, probability known, probability known. Because of AND gate 2 and 3 it is multiplied, because of here also AND gate these 3 multiplied, here also AND gates these 3 multiplied. Where we are assuming that basically there independent, independent that basic events are independent here, then it is and OR gate. So, what happen input? 1, 2, 3 and 4 is the OR gate formula and you will be ultimately getting this probability. So, leakage of gas, it is probability is 0.0001, ok ao, 1 in 10,000, 10 to the power 4 of 4 times you use the oven the kitchen suppose 10,000 times, you have used this and 1 time there will be a leakage.

(Refer Slide Time: 30:06)



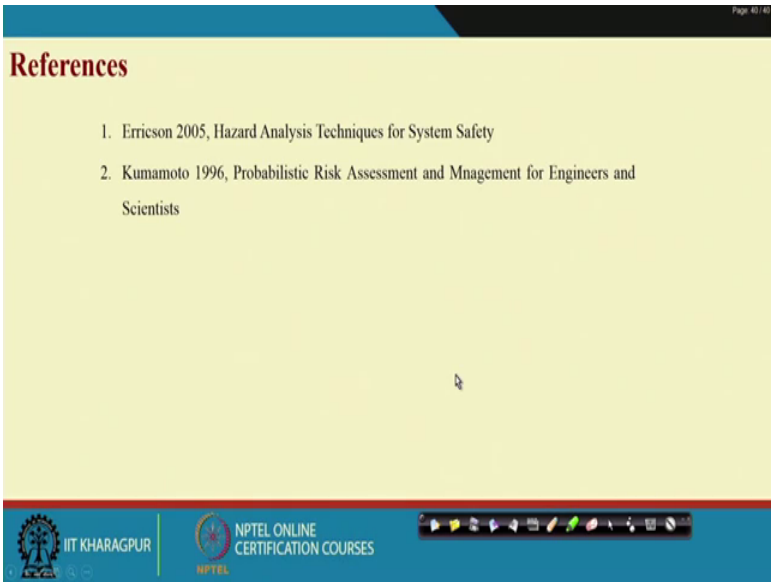
MISCUP algorithm just reverse. So, if for the leakage of gas top event we started from gate one and proceed through gate 5 in the MOCUS case. In MISCUP will start from G 5 the lowest gate and proceed up or so. G 5 is OR gate and the though 2 principal; that means, increase in element in a cut set or increase in number of cut set that principal will be applied it. So, G 5 is OR gate then G 4, G 4 is an AND gate that is why number of elements increases. Now G 4, 4 6 5 in this manner you proceed then finally, ultimately you will get the same.

(Refer Slide Time: 30:51)



Same number of minimal cut set.

(Refer Slide Time: 30:59)



So, I hope you got it. If you have any query regarding cut set method, please use the forum.

Thank you very much.