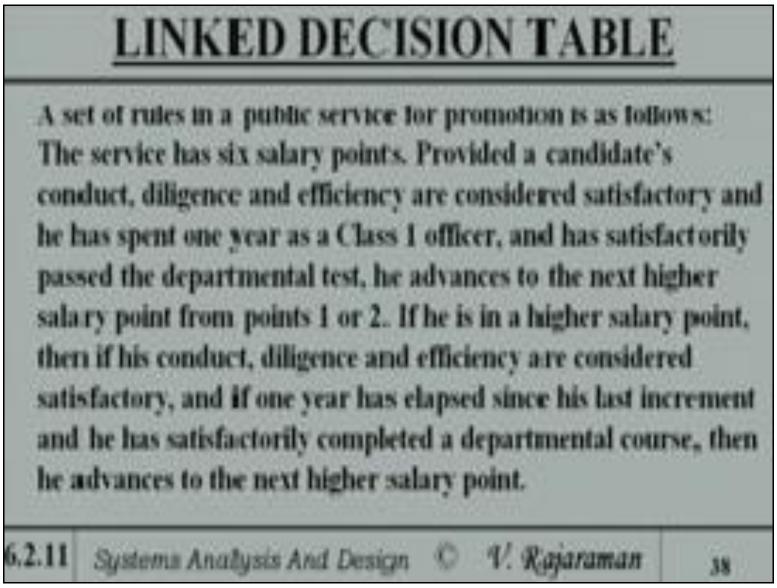
System Analysis and Design
Prof. V. Rajaraman
Department of Super Computer Education and Research
Indian Institute of Science, Bangalore

Lecture - 18

Last time, we were looking at length decision tables. And what I told you was to look at the table and try to interpret it. Now, I will do it slightly differently. I will give the set of rules and from that, I will go to the link decision tables. To explain, how the linking is being done.

(Refer Slide Time: 01:42)



Now, the table corresponds to particular word statement or a set of rules given by a government office. Some mythical government office does not exist in reality. So, the promotion rules are as follows. This particular service as 6 salary points. What is meant by salary points is that, given salary point, you get a certain amount of salary. When you advance to the next salary point, you get some increment, extra salary. And the number of different salary points.

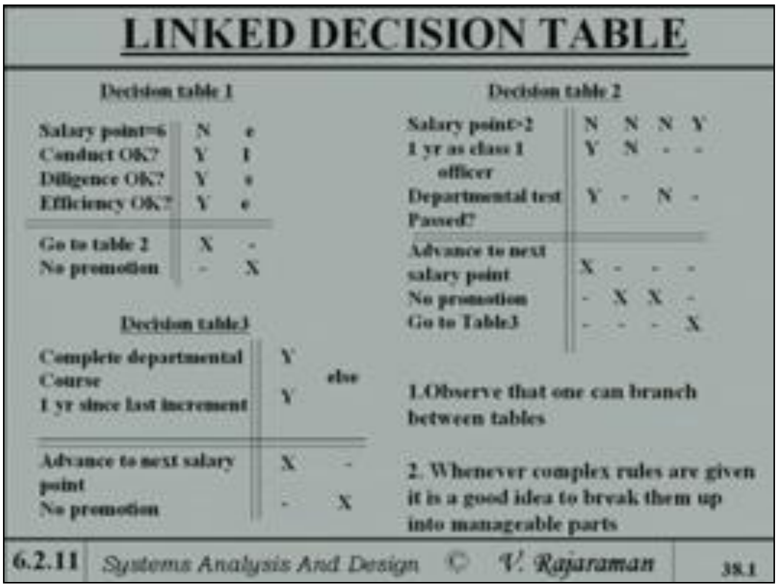
And each salary point one gets converted from one to the other. There is some certain conditions to be fulfilled by the candidate, who is up for promotion. Provided the candidate's conduct diligence and efficiency are considered satisfactory. That means these, these three are conditions which are essential to satisfy. That means, all three of them have to satisfy. If any one of them is failed, then I think the rows are not applied. And he has spent 1 year as class one officer.

And has satisfactorily passed the departmental test. He advances to the next salary point; if it is a given point 1 or point 2. That means, what it says is that there are in the beginning it says there are 6 salary points. For salary point 1 and 2, 1 or 2 there are that is the rule. And it is in a higher salary point, that is three or above. Then, if his conduct diligence and efficiency are considered satisfactory. And a 1 year has elapsed, since his last increment and he has satisfactorily completed a departmental test, departmental course.

Then he advances to the next higher salary point. The first one, talks about test to be passed as ((Refer Time: 03:49)) salary 1 or point 1 or 2. In this case, it is a course to be completed. Then only he advances to the next salary, higher salary point. It is obvious, that there are only 6 salary points. So, once you reach the 6 salary points, you kind of saturate. There is no further promotion possibility. Because, there is no such thing as 7 salary point. So, the first thing you do, look at the number of conditions in this set of rules.

The conditions are conduct diligence efficiency salary point. Whether, he is a suspend 1 year as a class one officer. Whether he has passed the departmental test and whether his salary point is above 2 and also, whether he has undergone a departmental course, completed a departmental course.

(Refer Slide Time: 04:55)



So, the total number of conditions, if you count all of them becomes, as shown in this table. Here the first plus 2 6 plus 3 7. Total of 7 conditions are there. You put all the 7 conditions in one table. The table will become extremely unreadable. That is the reason why, we breakup. The seven conditions means, there are possible 2 to the 7 rules. And 2 to the 7 rules means that is 128 different rules. And that is extremely difficult to one after the other consider and so on.

So, it is much better to kind of look back up the problem into smaller parts. So, that each part is somewhat, self contained and that part is links, link to the next set of decisions to be taken. So, the first is the way I have broken it up. The breaking up is not, there is no unique way of breaking up. I have particularly used a particular way of breaking up. Which, I consider a logical method of breaking up. You may consider something else, but you can try it out.

But, my feeling is that this reasonably alright. In terms of the way in which, I have done it. So, if the conduct diligence and efficiency all three are important. So, I put all of them in a in the same table. And we know that, if all of them are satisfied. Then only, you have to go into the rest of the promotion rules. Or go into the next salary point and so on. If none of them is actually, any one of them not satisfied. Then you do not have to really examine further, which salary point is and things like that, in terms of the rest of the rules.

Now similarly, when a person as a salary point 6 and if the answer is yes to that then there is the it means that the, if he is not in salary point 6. Then only, all these things apply. If he is in salary point 6, then that is the highest salary point. Beyond that, there is no promotion possibility. So, we are looking at the fact, that the, if he is salary point, which is 6 or below. That is it is not 6, but that means, it is below 6, because there are only 6 possibilities 1, 2, 3, 4, 5.

I could have put salary point less than or equal to 6. But it is alright. I mean, in fact it may be more appropriate to put less than or equal to 6 node, it means that it is a 6 or above. If it is 6 above as no meaning that is the reason I put equal. Because, there is nothing called 7 salary point. So, if the answer to the first question is no and all the others are satisfied. That is this conduct diligence and efficiency are all ok. Then you proceed further, to check various other conditions.

To decide where to advance to the next salary point or not. Otherwise, you say no promotion. Because, if it is in salary point 6 already. Then obviously, no promotion. If any of the conduct diligence or efficiency is not ok. Obviously, he is not going to be promoted. So, now I have put a cross against go to table 2. What it means is that, you have to examine table 2 now. If the first rule is satisfied, so the now go to salary the table number 2.

There I will look at all cases where, the salary point is greater than 2. Because, there is a particular rule for 1 and 2, and there is another rule for greater than 2. So, I am taking salary point greater than 2 now. That means, it is either 1 or 2. So, if it is either 1 or 2 or 1 year as a class one officer. He has finished 1 year as a class one officer and he has passed departmental test. Then he advances to the next salary point that is what the rule says.

So, if the salary point is not greater than 2 that means, it is salary 1 or 2. And he has not finished 1 year. As a class one officer then there is no promotion. Similarly, if he has not passed the departmental test, then also there is no promotion. If the salary point is greater than 2, that means if it is 3, 4 or 5. Then you go to table number 3, which is the next set of rules. You go to table number 3 that means, we can beyond salary point 2, you have to complete departmental course.

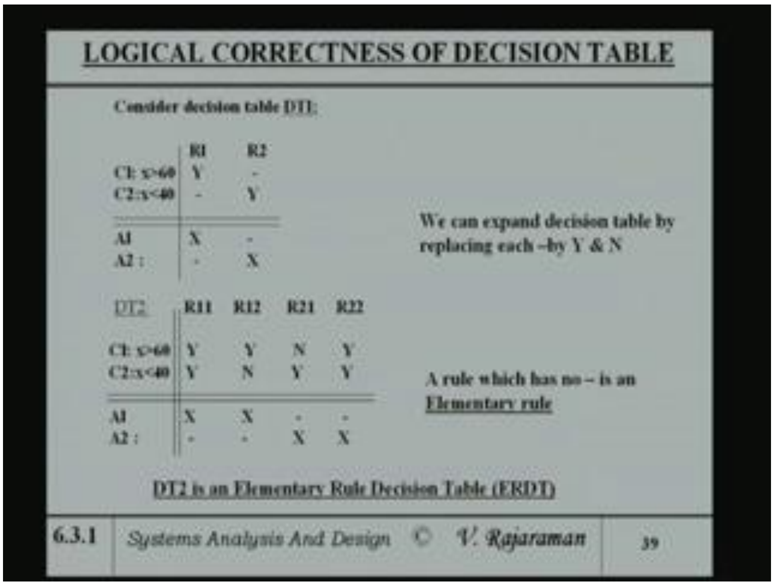
So, decision table 3 says, because the last one. That is salary point is greater than 2, you go to the table 3. The other case, the first rule obviously says, you have to go to the next salary point. Because, you have satisfied the conditions, specified by that rule. Then the next salary point once you go there, you have to find out whether departmental course has been attended. He has successfully attended departmental course and one year has been completed till the last increment.

In other words, you do not advance anybody to next salary point as per the rules. If not even one year is over, since he got the last promotion. So, he has completed one year since the last increment. And completed the course, then you can take him to the next salary point, else there is no promotion. So, observe that one can branch between tables. And, whenever a complex rules are given, it is a good idea or good practice to break it up into set of manageable parts.

So In fact, the advantage of this is that this also can be follow very easily by the manager, who gave you the set of rules. They may look at these rules he can kind of convert that, like I tried to do last time, back to the word statement. And see if it actually is what he intended as rules. Or, there is some error and he may like to change the policy and so on.

So, the whole idea is to be able to also be a very good communication made to the user apart from the person, who is going to take this table. And, who further work on it, which will be primarily to interpret it and write a equivalent computer program for it and so on. And that is exactly the idea of breaking up.

(Refer Slide Time: 12:38)



The I do not now like to look start looking at a very, very important part of the decision table terminology, namely at the end of developing a decision table. You would like to find out, whether the table is correct. That means, there is no logical errors in it. And you would like to also find out, that it is complete that means, no rules are left out. And you also would like to find out, if there are redundant rules. That is rules which are not necessary which can be eliminated. So, the rest of these lectures I am going to be devoting, in fact, rest of this lecture particularly.

To be i will be devoting to defining certain terms to find out the logical correctness of decision table. The point is that the lot of this is not really referred to in most text books. And i have done a fair amount of work in this and it is really based on some a lot of work done by me and my students. In terms of trying to find out, how to exactly determine the correctness of tables.

So, that is one of the greatest advantages of a tabular structure is to some extent, if there are small tables by inspection determine the correctness incorrectness. And also reduce any extra rules, which are there or any contradictions are there so on so forth. And if it is a very large problem with a lot of link decision tables in fact, we can even write a computer program to automate this testing process. In other words, the methodology I am going to say is am able to writing an algorithm and later on converting the algorithm to a computer program.

That means, in other words some of the aspects of this, which are syntactic some extent and not entirely semantic can be automatically checked. And, there are certain cases where even the semantics, can be checked with a by a program. I will kind of indicate those, those types of tables. Because, not every table is am able to that kind of a checking. I am going to essentially tell you later on, what types of tables are am able to automatic checking by a computer program.

Another advantage of table is that you can generate test data easily. Because, if you are able to generate data which will satisfy all the rules, one after the other. Then if you test these data one for then when you test using this test data. The actions were taken by the program, can be compared against the actions which are specified in the original specification. And, if they are the same, then you say that the test as passed. Otherwise you say test as failed.

So, the generating the test data is easy because it is based on the values, which assign to the conditions which are there in the table. (Refer Slide Time: 16:37). Now, we look at this condition this table. There are this table is as got two conditions. C 1 is x greater than 60 and C 2 is x less than 40. And the rules say that, if C 1 that is if x is greater than for 60. Then the action A 1 and if x is less than 40 do action A 2, that is what it says.

And it puts dashes to indicate that, in case number one rule number 1, you do not have to test condition C 2. And rule number 2, you do not have to test condition C 1. So, now this is exactly what the sometimes, when people interpret rules into tables they put a dash against irrelevant condition testing, which are done. In the last, in fact in the last table ((Refer Time: 17:40)) I have put a last of dashes.

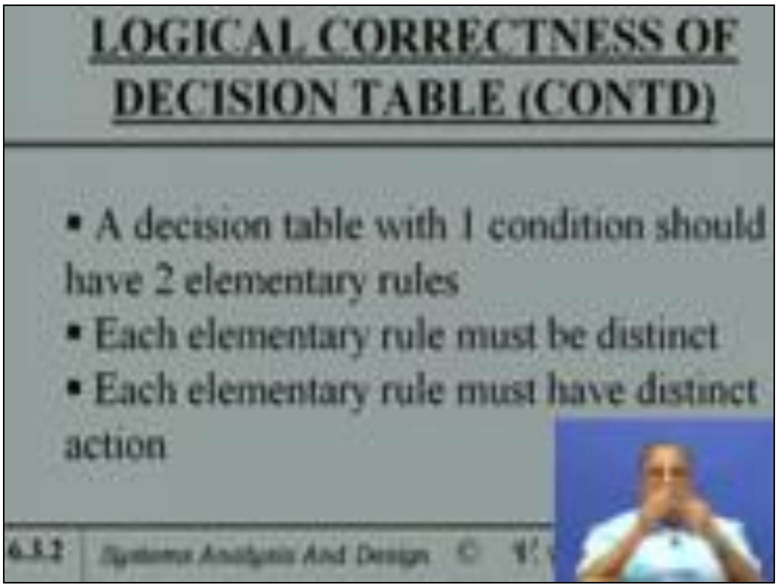
Because, in the second rule the, depart if he is the departmental test, whether he is passed or failed is irrelevant. Because he is going to get no promotion, because he is in you know, he has not completed one year as class one officer. So, ((Refer Time: 18:03)) dash is put somewhat from the word statement automatically by the by the person who develops it. So sometimes, dashes which you put can be misleading and so that is the reason why you would like to be able to expand out. And see, whether the dashes hide some information which is intended, so in this case for instance, I will expand the dashes until yes and no.

So, first rule will expand into R 1 1 and R 1 2, that means yes, yes and yes no. And similarly, R 2 will expand into R 2 1 and R 2 2, where C 1 is now the dashes expanded into no and yes. So, we can now see, there are actually 4 rules. Because, each dash expands into two possible rules, because yes or no, if there are two dashes in the same column it expand into 4 rules. So the, this case the expanded table is called elementary rule decision table.

An elementary rule is defined as one, where every condition entry as got a assigned value namely yes or no, in this case Boolean value yes or no. So, the elementary rule decision table is one, which is very nice to use or to kind of determine a number of things, which we were talking about. Namely, the missing rules, rules which are redundant and rules which are contradictory and so on. All these things are come out, once you expand a table with dashes into set of element into an equivalent elementary rule decision table.

That is you can say that the first table and second table are equivalent. In the sense that I have not done anything except put yes no for dashes. So, I have not changed anything else. So, purely syntactic conversion I might say. So, there is a, this conversion is somewhat automated. You might use, you might you will say.

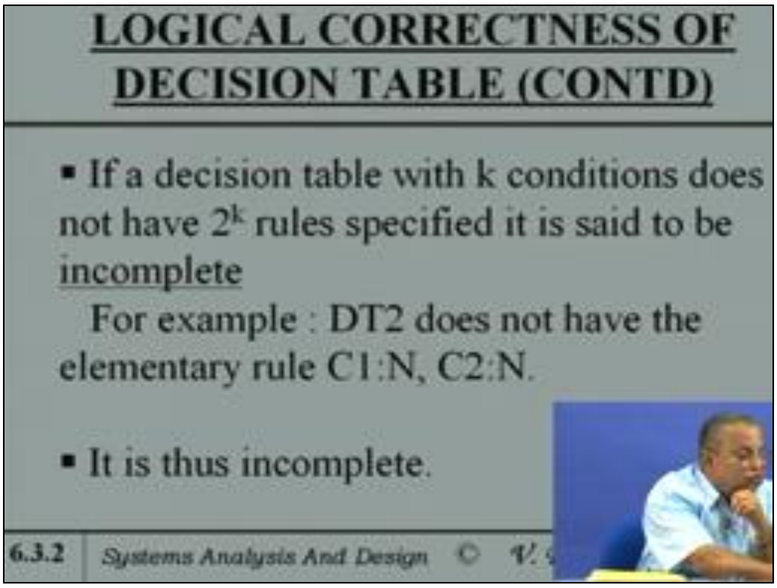
(Refer Slide Time: 20:31)



A decision table with one condition should have two elementary rules obviously, because yes or no. There is a decision table two conditions there should be total two or four elementary rules. Like we had in the previous in this one ((Refer Time: 20:48)) there are four elementary rules because there are two conditions. The three conditions requires two cubed or there should be eight, eight possible elementary rules.

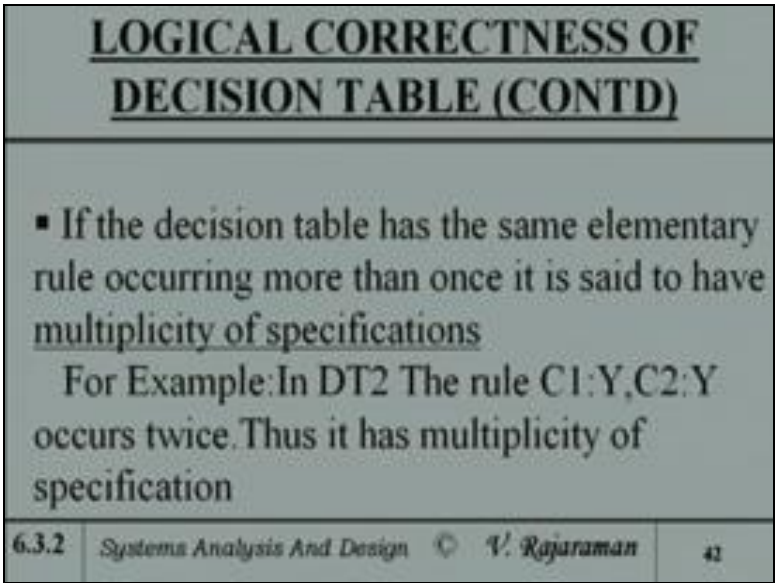
In general, if there are k conditions ((Refer Time: 21:02)) two to the k. Each element rule must be distinct. Because, it is the, the rule has got yes no yes no and it you know in other words the rules are all got to be distinct. And each elementary rule must have a distinct action. Like ((Refer Time: 21:27)) in this case each elementary rule. See now you can say, that there are distinct, but you can say R 1 1 and R 2 2 are really not distinct. Over the same answers are there and the actions are different. So, that there is some problem with this table, that is what it means.

(Refer Slide Time: 21:49)



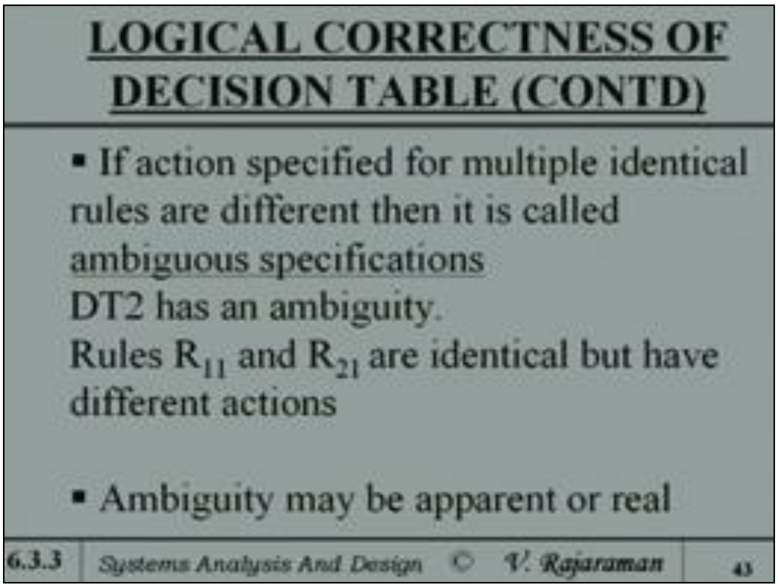
The decision table with k conditions does not have 2 to the k rules specified it supposed to be incomplete, because 2 to the 3 rules or 3 conditions expand to 8. 2 k conditions expand to 2 to the k. For example, digit 2 does not have elementary rule, in other words if i go back to this table ((Refer Time: 22:09)). The all the 2 to the k that means, 2 to the k have yes yes, yes no, no yes, no no. Now, no no is missing, so that means, this table has a missing elementary rule, so it is called incomplete.

(Refer Slide Time: 22:33)



If the decision table has the same elementary rule occurring more than once. Like in the table which I showed, it is said to have multiplicity of specifications. If were two yes yes answers in this case, there are ((Refer Time: 22:48)) two rules which are identical. And the multiplicity of the rules are there and the C 1 equal to Y, C 2 is Y occurs twice. Thus, it has got multiplicity of specifications.

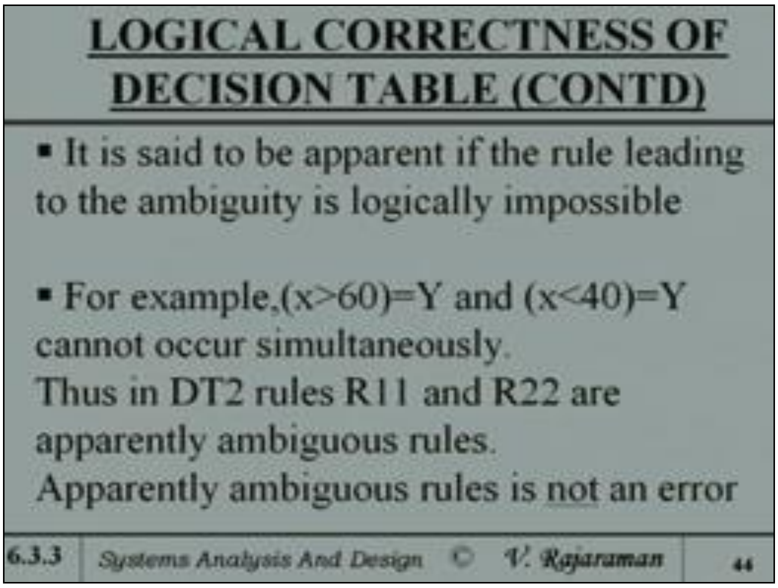
(Refer Slide Time: 23:10)



If action specified for multiple identical rules are different, then it is called ambiguous specifications. The specification is ambiguous DT2 has ambiguity. That see the because if I look at this ((Refer Time: 23:30)) as I pointed out. Rule number 1 1 has got an action A 1. And rule number 2 2 has got an action A 2. These two actions are different actions. So that means, it as got an ambiguity.

Because, R 1 1 and R 2 2 are identical, but have different actions. Ambiguity may be apparent or real, that may explain what is meant by apparent or real.

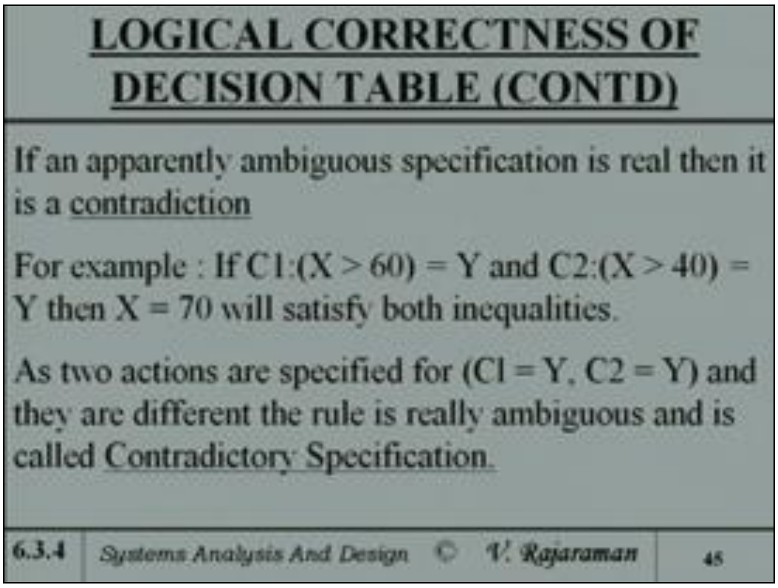
(Refer Slide Time: 24:03)



An ambiguity is said to be apparent, if the rules leading to the ambiguity is logically impossible. In other words, if that particular rule is logically impossible then the ambiguity cannot arise in practice. For example, yes yes answer is it possible or not. Now, x greater than 60 and x less than 40 cannot occur simultaneously. Because, i cannot find a number, which says above 60 and the same time below 40. So that means, this yes yes answer is not going to arrive it all and practice.

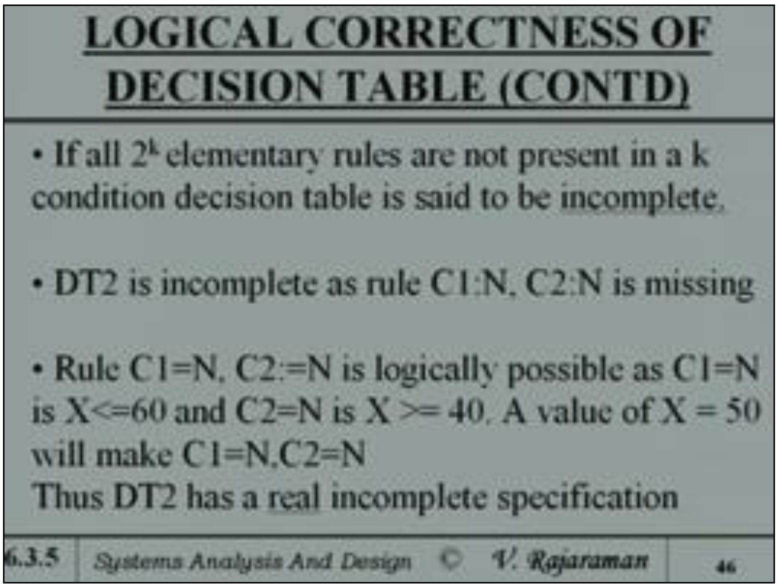
So, rule 1 1 and rule 2 2, so there is an actual. In this case then, these two rules there is ambiguity in the table is now real ambiguity, but an apparent ambiguity. So, apparently ambiguous rules are is not an error. Because, the fact that it cannot occur so question of an error does not arise.

(Refer Slide Time: 25:06)



If an apparently ambiguous specification is real that means, it can actually occur then there is a contradiction. Like for instance, if x greater than sixty is yes and x greater than 40 is yes. Then x is equal to 70 will satisfy inequalities. So, it means yes answered the first one and in this case C 2 and is ((Refer Time: 25:49)) you can say that the less than 40 greater than 60. But, i am looking at the other case.

(Refer Slide Time: 26:01)



So if the two to the k elements are not present then it is incomplete. D T two is incomplete, because there is no no. There is no specification of no no. It is a missing one.

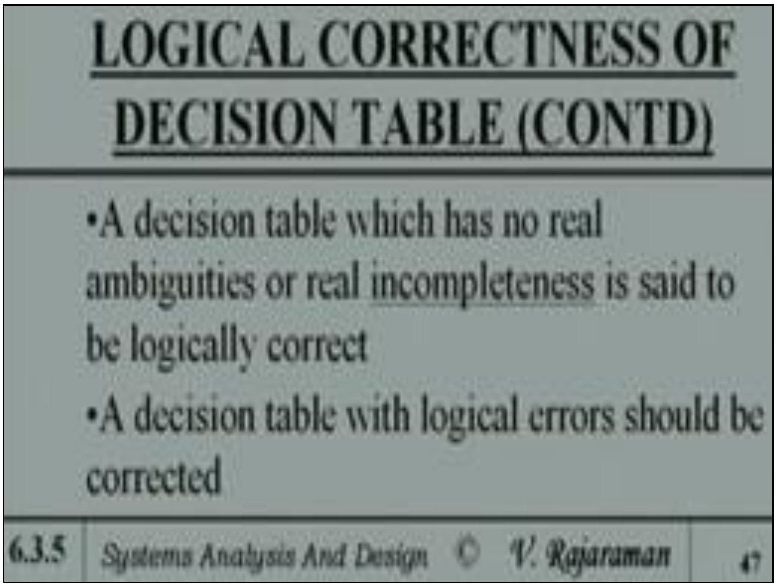
Like in this case, you can you count the ((Refer Slide Time: 26:23)) ones and see, you find that the no no, no no is absent. Because, yes yes is there, yes no is there, no yes is there, yes yes is there, no no is absent and so it really means that, for that if that particular rule arises.

If there is x less than equal to sixty and x greater than or equal to forty, that case there is no rule specified. ((Refer Time: 27:03)) it is incomplete because ((Refer Time: 27:07)) is missing. And question ask ((Refer Time: 27:11)) if it is really missing, is it a real kind of missing situation or an apparent missing. In other words, if it cannot arise in practice, the fact that it is missing is irrelevant. But, in practice if it is, if that particular set of conditions are satisfiable.

Then, obviously the no rule no action is specified. So that means, it incomplete are not given, an appropriate action for it. So, C 1 equal to N and C 2 equal to N is logically possible. Because, x less than equal to 60. And x greater than or equal to 40 is possible. That is a value x equal to 50, which will make C 1 equal to N and C 2 equal to N. So, 50 is less than equal to 60 and also greater than or equal to 40. So, thus DT2 has a real incomplete specification.

So even though, this particular decision table looks very ((Refer Time: 28:16)) very simple. It has a hidden incompleteness. And the incompleteness specific incompleteness is found out and also we find out the incompleteness can actually occur. So, any test data which you give, If you give 50 as test data for the table and try to test it with 50. It will come back and say no action can be taken. That means there is no rule specified for that and actually occurred. Because, there is a data value which does satisfy this.

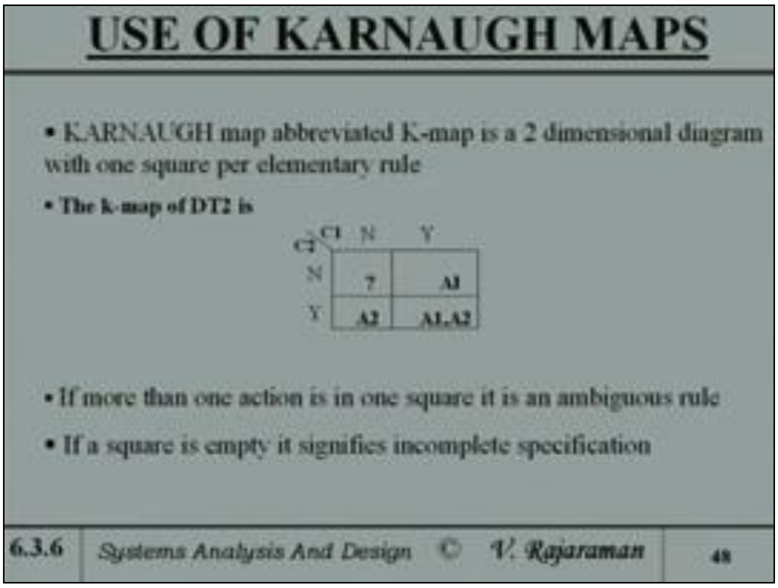
(Refer Slide Time: 28:54)



A decision table, which has no real ambiguities, or real incompleteness is said to be logically correct. That is it has got no, it is does not have any incompleteness. And, it is does not have any ambiguity, that is and both are real. That is there exist data which will satisfy the conditions, which will you know the, which if the data exist to satisfy the conditions. And the rule is either missing or it is duplicated with different actions, then it is an error. Otherwise, there is no error.

In other words, does not exist any data, which you know the point is if the data exist which will actually satisfy the inequalities and you get to a contradiction then of course, it is an error. If data does not exist it is not an error. And so, a decision table which has no real ambiguities. That means, as it has got no real you know no real incompleteness then it is logically correct. Otherwise, decision table with logical errors should be corrected, otherwise it has got incorrectness. Logical table with logical errors should be corrected obviously.

(Refer Slide Time: 30:25)



In fact, one easy way of finding out the missing rules as well as ambiguous specifications, purely syntactically. Is by mapping the conditions onto a map known as a Karnaugh map. You actually, would have come across Karnaugh maps in your course on logic design of computers. In fact, the first course on logical organization or the first course in digital design, which covers Boolean algebra and also talks about the truth tables and converting truth tables into equivalent switching circuits and so on.

Also, uses the same idea of a Karnaugh map. Karnaugh map is one which essentially two dimensional structure, which also in equivalent of the in this case decision table in fact you can look at the decision table, somewhat analogous to a truth table, a truth table with multiple outputs. Normally, in a truth table you have a set of independent variables and one dependent variable. There are some truth tables, where you can have multiple independent variables and a set of dependent variables.

But, in truth table the dependent variables are independent to one another. In other words, the within themselves the dependent variables are independent. And so the they also goes in a sequencing does not arise in that case is purely a syntactic equivalent. Whereas, in this case there is semantic involved, so d t is not the same as truth table. Where simple reason the actions are actions which are carried out and they have certain sequence to be carried out. So, it is not like a multiple output multiple input truth table.

So, there even though they the look similar. They are somewhat logically they have different purposes. And they are used in two different fields for one in the field of logic design to design switching circuits. And in the field of systems analysis and design to be able to analyze word statements and to kind of program those word statements. And also use it as a communication aid between the user and the program developer.

So, but the point is many of the tools and techniques, which you have learned in logical design become again useful in review of decision table or examining decision tables. So to some extent it is not something very, very new to you. It is something which you to some extent already know. So, I will go a little bit faster in terms of saying about Karnaugh maps and how to expand Karnaugh maps and so on.

And a Karnaugh map is named after person, who called Karnaugh, who first developed it or invented it you might say and ((Refer Time: 33:57)) abbreviated as K map. Two dimensional diagram and it also lists all the conditions. And all the actions in the case of Boolean truth table.

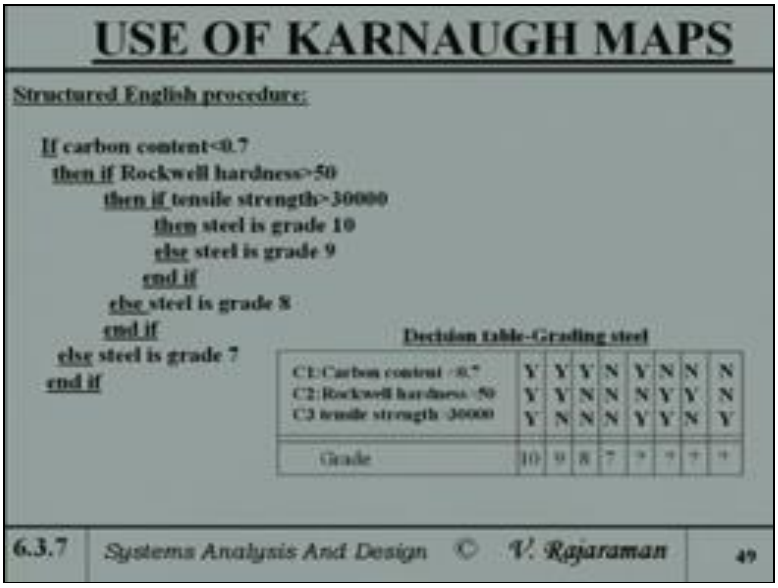
The 1s and 0s are used for C 1, C 2 and normally one table is written out for each independent variable. Whereas, in this case I am using the same tables, same Karnaugh map to be able to also show, the different actions to be taken. So, that it is it becomes, in other words I am mapping the table onto a Karnaugh map, which reflects whatever information's contained in the table d t is now contained in the Karnaugh map.

The Karnaugh map has now has got two conditions C 1, C 2 you have no yes along the top row and value of C 2 are no yes along the column. So, if you take no, no I will found out for no, no, no action is specified. So, you put a question mark, for a yes no C 1 yes, C 2 no there is action A 1 which is specified. If I go back I can actually see this ((Refer Time: 35:23)).

So, I am actually what I am doing is to map this. So, A 1 if it is C 1 is yes and C 2 is no, then A 1. If C 2 is yes ((Refer Time: 35:39)) A 1. And then, exactly what I am taking the same table and putting it in the mapped form. So, I essentially put that table into this map and I said you know once you cover it put these things, you can immediately see that yes, yes has got the problem.

Because, there are two actions specified. If more than one action is in one square, it is an ambiguous rule. Because, there are two different two actions are specified for that. If a square is empty, it signifies an incomplete specification. So, the advantage of a map is you can, once you map the table into the map, it more or less becomes obvious. What is missing and what is what is ambiguous?

(Refer Slide Time: 36:33)



Now, let me take another example of taking a the point I am the reason. Why I am putting this example is that, if you specify certain thing s in a structured English, it looks alright. But, with when you map it on to a dressing table. And also to a Karnaugh map, which is equivalent to that. If I find out that it may be incomplete.

Like for instance a structured English procedure is given. If carbon steel is checked the whole idea is a… This is a set of rules given by a metallurgical engineer about classifying steel. And the rules which are given is that, if the carbon content the steel is less than 0.7. And if the Rockwell hardness, there is something called Rockwell hardness is a measure for steel.

It is greater than 50 and if the tensile strength of this particular steel is greater than 30,000. Say the carbon content Rockwell hardness and tensile strength or properties of the steel is being manufactured. So, you check these three. That is, if carbon content is less than 0.7 and Rockwell hardness is greater than 50. And if the stris tensile strength is greater than 3000, then steel is grade 10.

If it is not tensile strength is not of strength 30,000, it is grade 9. If then the nested if says that, if the Rockwell hardness is greater than 50. But, if the carbon content is less than 0.7, if the hardness is not greater than 50, then steel is grade 8. If none of them see carbon content is less than 0.7 at none of these conditions are satisfied ((Refer Time: 38:48)) the steel is graded 7.

So, this is essentially what the structured English statement says. If I put it into a form of a table I have got three conditions. Carbon content less than 0.7 Rockwell hardness greater than 50. And tensile strength greater than 30,000. You see that first of all see interpreting a nested, if, then, else statement which is given as a structured English, it is not straight forward. In fact, when I tried to do it I found that I got a properly bracket and ((Refer Time: 39:32)). And I could easily made a mistake in incorrect bracketing. That fortunately will not arise in the case of a table.

Because, I am modules commountariley looking are all possibilities. And if there are three conditions and each condition can yes or no answer. The possible of total of eight possible rules two, two cube rules. So, the first rule says if all three are satisfied all three conditions satisfied. Then, the grade is 10, the first two are satisfied and third is not satisfied then grade is 9.

If the first is satisfied and the other two are not satisfied grade is 8. If ((Refer Time: 40:19)) is satisfied grade is 7, that is what the rule says. Now, what if the first one is satisfied and the last one is satisfied. That is carbon content is less than 0.7. And tensile strength is also greater than 30,000. But, Rockwell hardness is not greater than 50 what do I do, do I classify it as seven or do I classify it as 9 or 8 or what are is there a new classification of 6 I had to put.

So, in other words all the other cases are not specified, in this particular business rules converted into a structured English statements. So, structured English statement sometimes can may misleading, it may look complete. But, it may not be complete in the sense that all possible contingencies may not be entirely covered, which we find in this case.

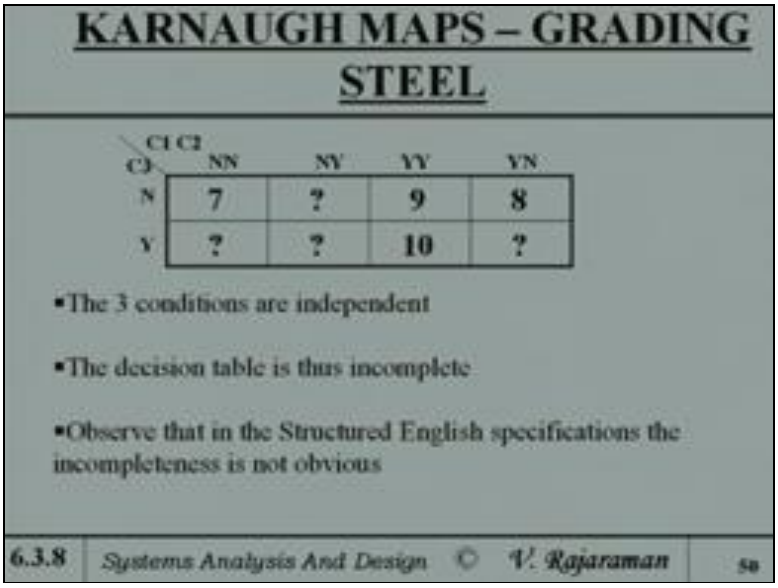
The four possible rules for which they it is not cover. It may happen and I have no idea. Because, I am not metallurgical engineer, that if carbon content is greater than 0.7 and tensile strength is greater than 30,000. If the hardness is less than 50 may not occur at all or even if it occurs it is rejected. So, if it is rejected then I must have one more rule saying that the carbon is the particular thing is not accepted or if I assume that grade seven is non acceptable grade.

Then of course, I can say that I can put seven for all of the rest of the things. So, the point is that the missing rules, it is not for the analyst to decide, what to put in that missing rules. It is for the user, in this case the metallurgist who has given these rules, to tell me whether these will be relevant or irrelevant. See, your steel it may not be correct at all, these points may be irrelevant, the question I am may be asking is stupid.

But, I do not know whether it is stupid question or not or whether he has missed it or not. So, it is important to be able to point out the fact that, you know when I look into a rules for these rules, there is no action specified I no classification to be given. So, what do I do, do I leave it unclassified or does not have ever occur in practice or do I give a new classification number and so on.

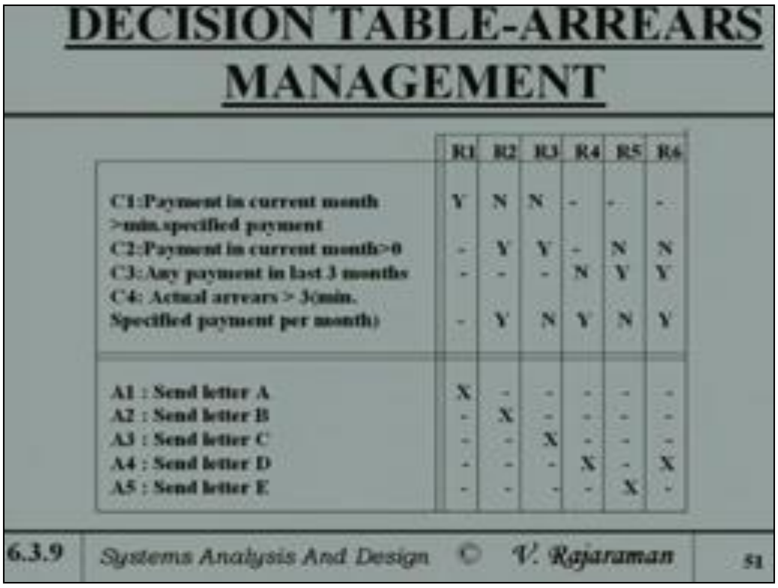
So, the point is that the decision maker is not you, decision maker about the action may be giving grade is user. So, you have to get back the user.

(Refer Slide Time: 43:34)



So, you can map it into a Karnaugh map it is quite obvious, where three conditions are independent and then there are question marks. So, decision table is incomplete and incompleteness is also maybe it looks like it is a real incompleteness. Because, in the absence of any information, it is a real incompleteness.

(Refer Slide Time: 44:05)



	R1	R2	R3	R4	R5	R6
C1:Payment in current month >min.specified payment	Y	N	N	-	-	-
C2:Payment in current month>0	-	Y	Y	-	N	N
C3:Any payment in last 3 months	-	-	-	N	Y	Y
C4: Actual arrears > 3(min. Specified payment per month)	-	Y	N	Y	N	Y
A1 : Send letter A	X	-	-	-	-	-
A2 : Send letter B	-	X	-	-	-	-
A3 : Send letter C	-	-	X	-	-	-
A4 : Send letter D	-	-	-	X	-	X
A5 : Send letter E	-	-	-	-	X	-

I am taking one more example of trying to analyze a given decision table to find out, whether it is got any errors in it. A set of this decision table is given for management is arrears in a particular company. The arrears management table, which is given as payment in current month that is there are four conditions to be checked for arrears management.

If the payment it may be that of a credit company you see, the credit company gives you cards and at the end of the month you are supposed to pay. And if you do not pay of course, I ((Refer Time: 45:03)) with interest. And if you do not continuously pay for a certain number of bands, they may cancel the card. And if you do not pay for one month they may send you some reminder.

So, different actions the company might take and they are called business rules taken by the company. So, in this case the payment of the current month is greater than a minimum specified payment. Normally, in a credit card company, you are supposed to pay each month. Even though you have ((Refer Time: 45:38)) something like, you may have purchased items up to 5000 rupees.

But, they will tell in statement that it is enough if you pay 200 rupees, the rest of it will be charged in terms of the interest will be charged in that. So, there are certain kind of minimum specified payment. So, payment of the current month is greater than the minimum specified payment. The answer is yes, ((Refer Time: 46:05)) letter a. That is you send an acknowledgement, letter thank you for your payment.

But, you still have an arrears of so and so. And that arrears will carry an interest of so and so forth. If it is these letters are normally generated automatically by computer. And sent out to the customer. Because, this is essentially a set of rules implemented and by actually computer program. The rules have be given by a manager, but ultimately the actions are automated. That is whole idea of using computers.

So, the first rule says the payment of the current month is greater than minimum specified payment, you send letter A. If the current month it is not greater than minimum specified payment. That means, he has given less than what was specified as the minimum payment due. That is as I said every credit card company has some minimum payment, which they allow.

If it is below minimum payment card is greater than equal to minimum as I said it is below minimum. And payment in the current month is greater than 0. That means, he has given some money, but he has not he has given money. Suppose, 500 rupees paid as what is required of him and he has paid partially 200 rupees, but not 500 rupees.

Then, payment of the current month is greater than 0. But, it is not full amount and the actual arrears is greater than 3 times minimum specified payment per month. That is, if there is a minimum specified payment for each month. If you take 3 months average and if it is greater than 3 months specified payment per month. Then, you send letter B giving him some kind of warning maybe saying that your arrear is building up.

And so interest payment will start accumulating. If the third rule says, that if the payment current month is greater than 0. So, I will say, but it is less than the what is required in the month. And an actual arrears is less than 3 months minimum specified, then you send another letter C. So, these are third possible rule. So, it may actual letter is for the manager to decide what letter he will be sent off with a machine.

And rule forces that first whether the first two are irrelevant. If no payment has been received any payment in the last 3 months no. That means, no payment has been received in the last 3 months. And actual arrears has exceeded three months minimum specified arrears per month. So, in fact you are accumulating date very, very fast.

And then, when you say send letter D. In the case of rule 5 payment in current month is greater than 60. You know ((Refer Time: 49:37)) 0 you know; that means, it is less no payment as been received. And any payment in last three 3, yes some payment as been received in last 3 months. And actual arrears is greater than 3 months, you know less than 3 months, then you send letter E.

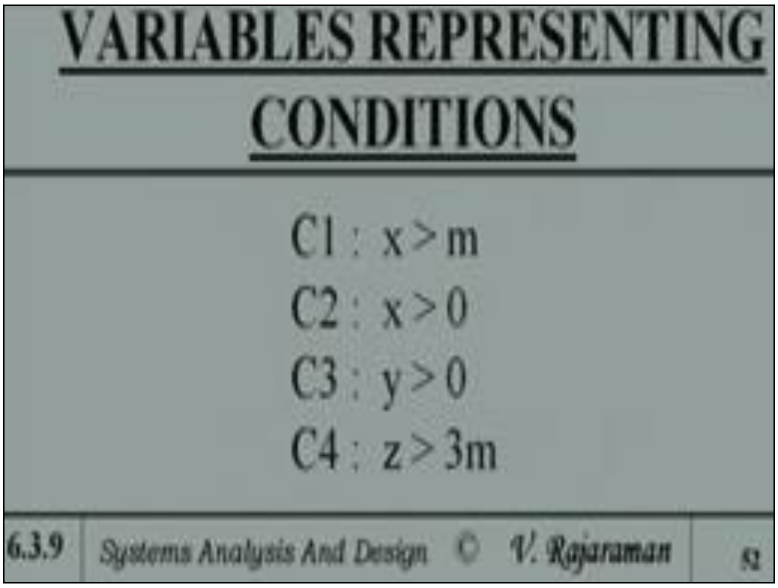
The last rule says that any payment in the current month is not greater than 0. That means, it is less than equal to 0. That; obviously, means 0 no payment. And any payment in the last 3 months, yes there is some payment. And actual arrears is greater than 3 months arrears. Then, you also send letter D there are two cases rule 4 and rule 6 you send letter D. And the other rules you send A, B, C, D, E.

So, this is some arbitrary I mean I just picked out from my heart. Now, the next question is how do I analyze this set of this decision table to find out whether rules, which are given are complete. Rules, which are given are not only complete. But, they do not have any the ambiguities and if there are ambiguities are they real ambiguities or are they apparent ambiguities.

That means, I really contradictions in the rules. These are the points we found out before accepting these rules, the implementing on the machine. Because, you will find out later on if you did not check it out, check out the DT carefully, you will find out later on that a particular case may arise, when the wrong letter would be sent. And then, the customer will be extremely unhappy or the machine will just hang up and say for this particular set of conditions I do not know what to do.

If it is incomplete specification. That means, the machine will hang up and say I do not know what to do. So, first thing we do before we analyze is to kind of convert this into a that is take each condition and use variables for that.

(Refer Slide Time: 52:06)



Because, it is easier to work with variables. Because, mathematically we can analyze it easier, if we use variables and inequalities. So, in this case I am using x for payment in the current month x and m is for the minimum specified payment. So, I have converted the payment for a current month is x and the minimum as M C 2 is payment in current month that is x it is greater than 0, it is second condition.

So, the second condition x greater than 0, third condition is independent that is any payment in last 3 months. Last 3 months payment is accumulate the payment in the current month. So, that is a new variable y. So, I put Y for C 3, C 3 is any payment in last 3 months is when Y, Y is greater than 0. And actual pay arrears is greater than 3 months minimum specified per month.

The actual arrears is some z which is a variable. Because, the arrears will change from person to person to person and it is not related to x and y. So, I put a new variable z and it is greater than 3 months m plus what was used for minimum specified payment. So, this says 3 months minimum specified payment per month.

So, I converted the table, the condition that the condition the condition stub into equivalent quantified inequalities based on the what I saying, if I inventing or putting variables instead of words. Instead of putting words I used mathematical variables, which represent the words. And the reason I have done that is obvious.

That is, I am effectively going to be able to analyze it, mathematically only if I do this first. So, that is exactly what you have to really do, given a table if you want to start looking at the checking the table for contradictions and so on. The first thing you have reduce to able to invent simple variables. And inequalities if inequalities or whatever based on the specified condition and use that for further analysis.

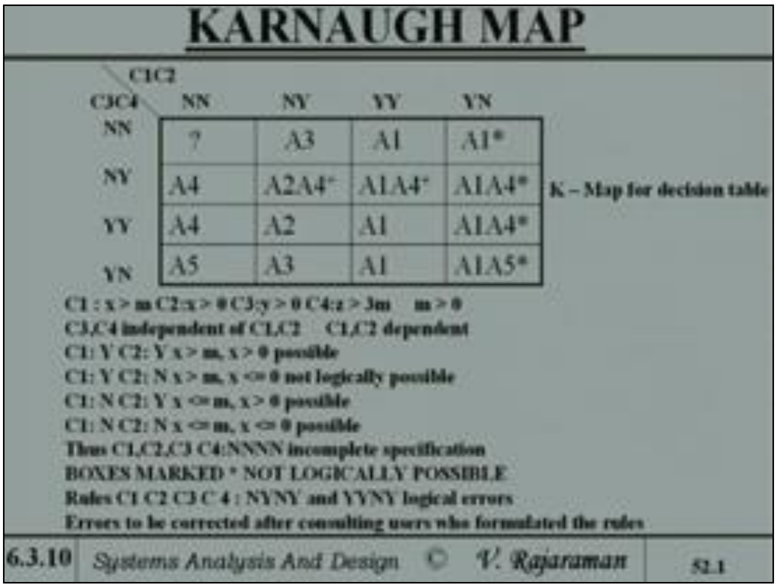
The further analysis then becomes fairly straight forward. So, what I have done is taken this table and mapped it into a Karnaugh map. Now, in fact the way in which to map it with the Karnaugh map is as I pointed out. Let us straight away we can take an example. Rule 1, rule 1 has got action A 1.

Rule 1 will expand into how many rules, because there are three dashes in it. Rule 1 will expand into total of 8 possible rules. So, rule 2 will expand into two possible rules. So, I add 8 plus 2, 10. Third rule expand into 2 plus 2 12. Fourth rule expand into 4 16. Next rule expand into 2 more 18 plus last rule expand into 2 20.

So, total of 20 rules this table will expand into total of 20 elementary rules. So, if I convert this into an elementary rule decision table, with only yes or no answers, there will be 20 elementary rules. Whereas, the total possible elementary rules with four conditions in a table are only 16 but there are four access rules.

And so; that means, there is some problem with this table to begin with, so it as some access rules. So, I will be essentially certain rules may map on to a map on to multiple actions. And I do not know by just counting, there is any missing rule or not. And that is what exactly the Karnaugh map does. It when you map it let us take each rule one by one. Let me just take the first rule and see what, which is action A 1.

(Refer Slide Time: 57:26)



First rule is yes for C 1, C 1 is yes and C 2 is C 1, C 2, C 3 can be anything. Now, if I look at C 1 is yes is last column and C 2 is irrelevant. So, A 1, A 1, A 1, A 1 all four come along there. So, four are there. So, I had A 4 more, second column is also C 1 equal to yes, C 1 is yes the second column. Now, again A 1, A 1, A 1 A 1.

So, the C 1 equal to Y will map onto 8 rules, which will essentially be the 8 squares in this map the 8 squares in the map are the last 2 columns in the map. The last 2 columns in the map, essentially allow all A 1. And if I similarly I go on, if I convert second one it is no Y, no Y dash Y. No Y dash Y will become no Y is C 1 C 2 is no Y dash Y becomes no Y and yes Y.

So, A 2 action A 2 maps onto the second row, second column and third row second column I put that A 2 down. Similarly, if I take the next rule namely ((Refer Time: 59:02)) no yes dashes. So, no yes dashes will become no yes is C 1, C 2 no yes and dash becomes either no yes. So, the first row first column, first row second column and in fact, I had to go along second column. Second column at the last row at all action A 3.

So, I mapped on to action A 3. And similarly, the ((Refer Time: 59:32)) mapped into four actions. In fact, I can see that the no yes for C 3, C 4. So, if I go along no yes of C 3 C 4 no yes is second row for C 3, C 4 and all A 4 will map on to that particular row. So, I continue like this, in other words I just expand the dashes into yes no and fill up this table.

Once, I fill up the table I map the stars, those particular. You know, in fact there are stars and pluses I will come to why I put pluses and stars in the next lecture. But, I put a question mark where is the missing rule for nothing is specified.

Even though there are 20 rules. But, if we count the total here 3 plus 4 7 plus 3 10 plus 3 13, 15, 17, 30, 30 rules are already there. So, that is where I stop now and continue next time with analyzing this further.