Manufacturing Systems Management

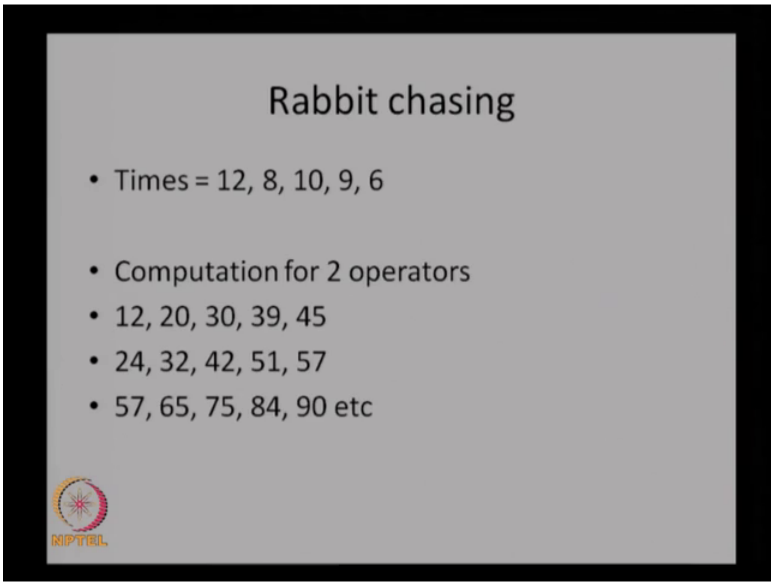
Prof. G. Srinivasan
Department of Management Studies
Indian Institute of Technology, Madras

Lecture – 21

Operator and task assignment

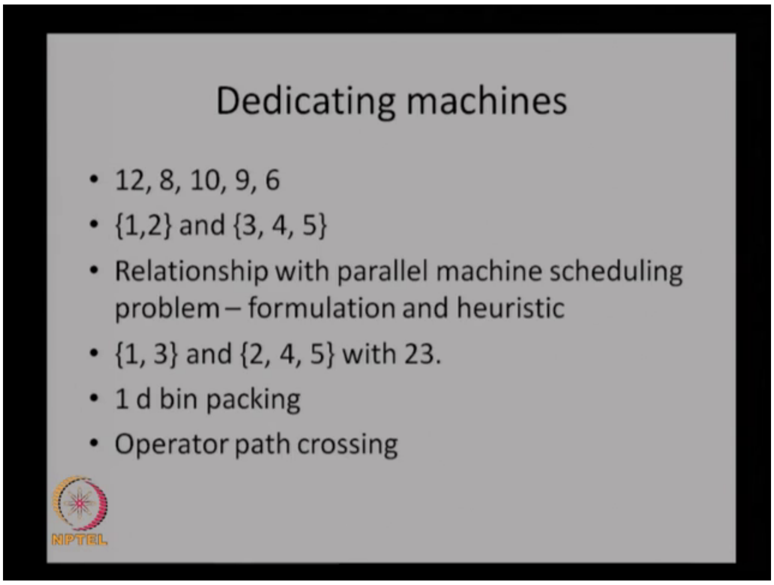
In this lecture, we continue the discussion on operator and task assignment. In the previous lectures, we have seen some aspects of operator and task assignment. We have seen some aspects of rabbit chasing and dedicating machines to operators.

(Refer Slide Time: 00:23)

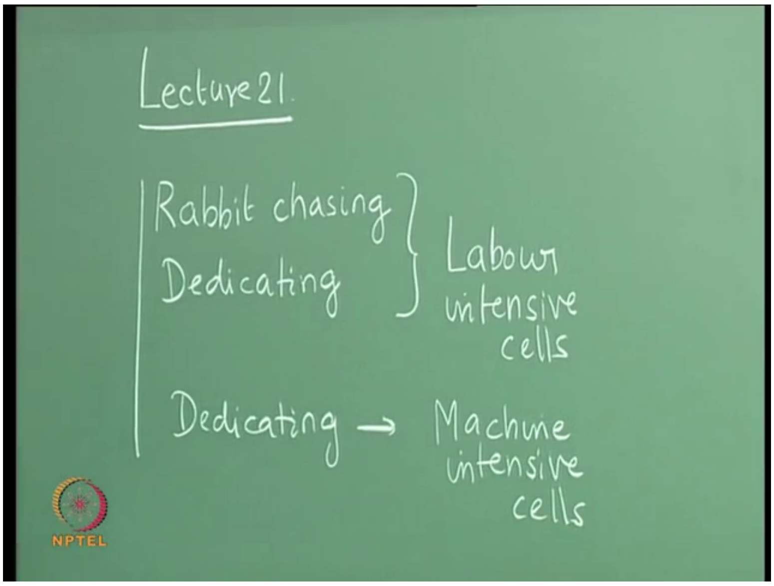


So, we have seen rabbit chasing and we have seen some aspects of dedicating machines to operators.

(Refer Slide Time: 00:26)



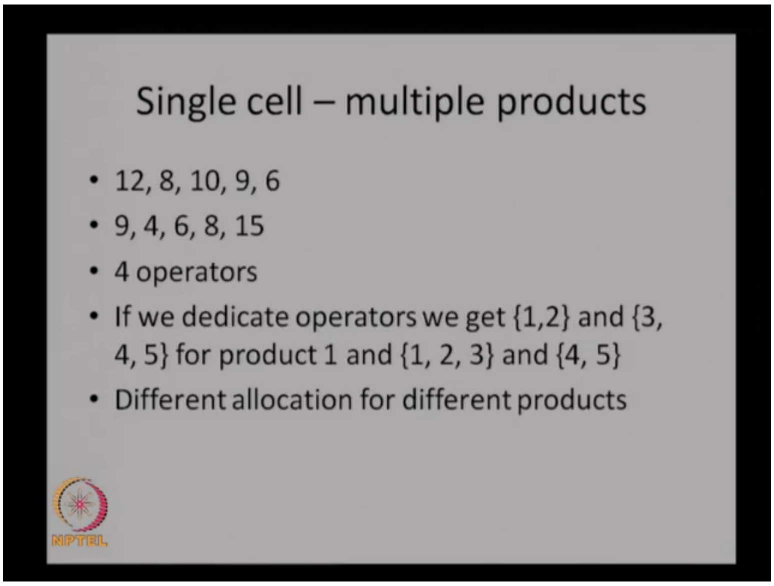
(Refer Slide Time: 00:35)



We have also seen dedicating both these are seen under the context of labor intensive cells and we have seen dedicating machines to operators in the context of machine intensive cells. In the context of machine intensive cells, we assume that while the operation is being carried out or the manufacturing takes place, the operator need not be there with the machine, which means the operator does only load and unload and while the actual manufacturing or machining is happening the operator can attend to some other machine and can load or unload parts on other machine.

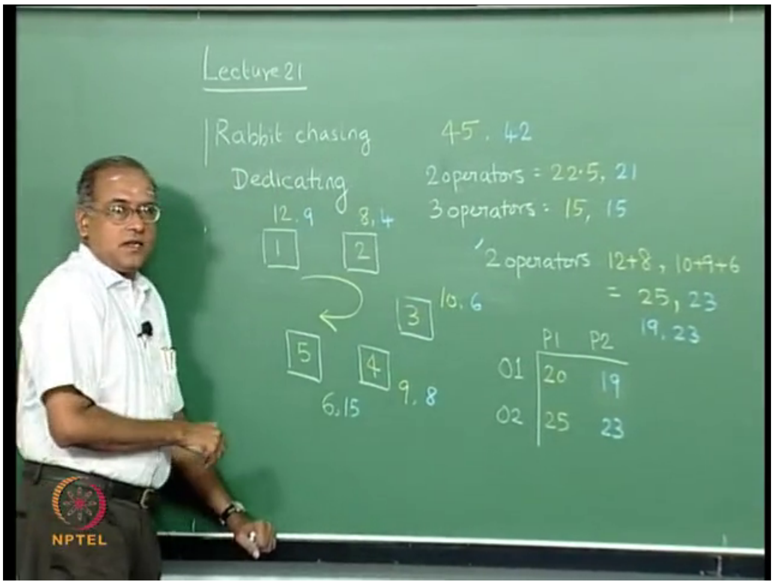
So far, we have seen 3 models basically, where one is the rabbit chasing model and the dedicating model, where in the context of labor intensive cell the operator is physically present when the actual machining takes place. We also saw that rabbit chasing gives better output and can increase the productivity, but has its own concerns such as limit on the distance that people walk, ability of the operate all the operators to handle all the machines and so on.

(Refer Slide Time: 02:40)



So, the next model that we will see is a model where we have a single cell and we do multiple products. So, we have a single cell and multiple products under the context of rabbit chasing and dedicating.

(Refer Slide Time: 02:54)



So, let us assume that there are 2 products this cell makes. This cell has 5 machines and this is the direction of the flow.

The process times are 12, 8, 10, 9 and 6 for the first product and the process times are 9, 4, 6, 8 and 15 for the second product. So, the sum of the processing times is 20, 30, 39 plus 6, 45 for the first product and 9 plus 4, 13, 19, 27 plus 15, 42 for the second product.

Now, if we have 2 operators, the first product we can expect a piece every 22.5 time units and the second one we can expect every 21 time units. If we have 3 operators, the first one we could expect 45 by 3, 15; 15 is more than 12, 8, 10, 9 and 6. So, we would expect 15. But, 42 divided by 3 is 14 and this 15 is more than 14, so the output will come every 15 minutes.

So, we could do this analysis for 2 operators, 3 operators and so on and if we do rabbit chasing, we can compute what is the expected output at steady state. Then we dedicate the machines to operators. We have already seen that the best output in this case for 2 operators would be 1 and 2 which gives us 20, 10, 19 plus 6, 25. So, 2 operators we would get 12 plus 8, 20 and 10 plus 9 plus 6, therefore, the best value is 25 is when is the steady state output of this system.

Now, when we take the second product, we have 9 plus 4, 13 plus 6, 19, 15 plus 8, 23. So, for the second product we would get 19 and 23. So, the output will be 23, for 2 operators. The only thing we understand here is, if we are having 2 operators and if we are dedicating machines to operators under the context of a labor intensive cell, then we observe that for the first product the operator 1 would do 12 and 8 which is 20, while operator 2 would take 25 time units; while for the second product, the first operator will do the first 3 machines will do 19 and the other operator would do 23.

So, we have to look at 2 things; one is that for different products when we dedicate we will have situations where operators will handle different machines or there is different machine allocation when multiple products are considered. This also brings us to the requirement that the operators should be able to handle all the machines. We have a situation where operator 1 would be doing only 1 and 2 and operator 2 would be doing 3 4 and 5, when product 1 is being made; while operator 1 will have to do 1, 2 and 3 when product 2 is made.

So, this necessitates that operator 1 should be capable of handling 1, 2 and 3 while operator 2 should be capable of handling 3, 4 and 5 when we have multiple products being made in the same cell. We can have another situation which necessitates that all operators should be able to handle all the machines.

Now, let us assume that we are doing product number 2. While we do product number 2 and we dedicate operators to machines, when we do product number 2 the first operator would handle 1, 2 and 3 to have a time of 19. The second operator would have 8 plus 15 which is 23 and the output would be every 23 time units.

Now, since operator 1 takes up the things first and then operator 2 does it and if we have a situation where the operator 1 takes a piece, finishes this, keeps it here, goes back, takes another piece, finishes, keeps here and operator 2 takes these 2 then, if operator 1 is busy all the time then there will be a buildup of inventory at this point. Because, operator 1 would have the raw material or entering material inventory here and if here we have an unlimited supply or if the entire production batch is available here, then operator 1 would be taking a piece, he would finish these 3, leave it here, go back take another piece and every 19 time units the pieces will get built up here.

Operator 2, would be requiring 23 time units for every piece and this buildup is going to be replenished at 1 piece per 23 time units. So, there will be an inventory buildup, if operator 1 continues to work all the time. We have already seen that we should not allow or we should not plan inventory buildup or excess inventory buildup in the context of modern manufacturing methods and therefore, the only way by which we can prevent inventory buildup from happening here is to control the amount of inventory that is entering here.

So, this principle is a fundamental principle in just in time manufacturing which we would be seeing later in this lecture series, but if we control the inventory here then this person will be walking on a piece every 19 minutes, while this person would take 23 minutes or time units while working on this piece.

If we control the inventory at this point what will happen is that, this operator whom we call operator 1 will be idle for 4 time units, while this operator would be working all the time. So, when we have a situation where while dedicating there is a difference between these 2 outputs or there is a difference between these 2 outputs, 20 and 25 which gives us 25 for product 1, which simply means; if we make a table we say operator 1, operator 2, product 1, product 2, operator 1, product 1 is 20 time units, operator 2 product 2 is 25 time units.

So, operator 2 operate operator 2 product 1 is 25 time units. So, steady state output is 25 time units. When we consider the second product 9 plus 4, 13 plus 6, 19; this is 23. So, steady state output is 23. If we permanently dedicate according to this then the first operator will be doing these 2 for product 1 and these 3 for product 1; second operator will do these 3 for product 1 and these 2 for product 2 and if in both the instances it happens that the first operator is spending less time than the second operator.

While in rabbit chasing, if we have 2 operators both of them are going to work about the same time and a steady state output will be 22.5 for each at steady state and 15 for each if we have 3 or 21 for both the operators if we have 2 operators for the second product. While rabbit chasing utilizes them equally, dedicating does not utilize them equally. If we want to utilize them equally, then we end up building inventory at some point.

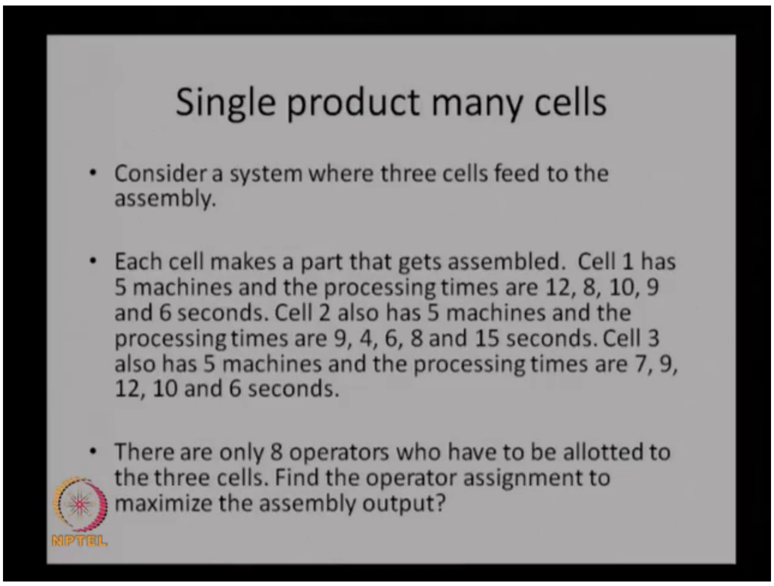
If we control the inventory then we would have less utilization of operator 1 and more utilization of operator 2. So, operator would be spending more time with the machines than operator 1. So, in order to balance that what we could do is, if the same 2 products are made let us say in 2 consecutive shifts, then the person who becomes operator 1 will be operator 1 in 1 shift and operator 2 in the second shift.

So, the people can change that the dedicating machines will remain the same. So, if we have a person called M and a woman called F which are our 2 operators; then M will be called operator 1 and say F is called operator 2. So, the person M in the first shift will be operator 1 and will be doing these 2 when the first product is being made and these 3 when the second product is being made. So, the person number 1 who will be M will now have utilizations of 20 and 19 in the first shift. While the person F who is now called operator 2, she will have utilizations of 25 and 23 in the first shift.

Now, when the second shift if we interchange them and the person F takes up these 2 for product 1 and these 3 for product 2 and so on. Now, this person will now the one who had 20 and 19 in the first shift will now have 25 and 23 as the time spent. So, that way in one shift one person will be utilized more the other will be utilized less. In the next shift this person who has utilized more will now be utilized less.

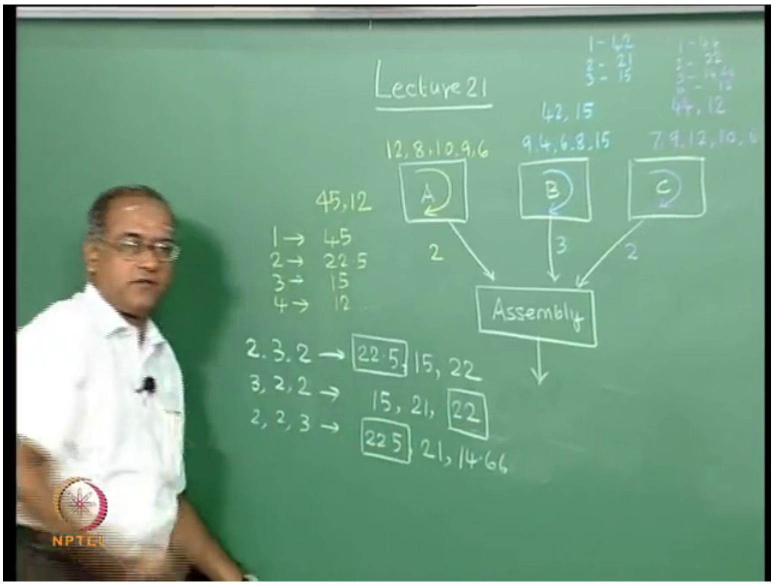
So, this way we could ensure that one person is not overworked in the cell, but if we have to do that again both M and F should be capable of handling all the 5 machines. So, both in rabbit chasing as well as in dedicating it is necessary that operators are multi skilled they are trained sufficiently so that, they can handle a variety of machines. So, operator training cross functional training is a very important aspect in the context of cellular manufacturing.

(Refer Slide Time: 16:19)



Now, let us continue further and look at single product with multiple cells. So, let us look at a situation where let us look at a situation where 3 cells feed to the assembly.

(Refer Slide Time: 16:45)



So, this will be called assembly. This is A, this is B and this is C, all of them feed to the assembly and the final product comes out of the assembly. Now, let us assume that each cell, cell 1 has 5 machines and the process times are 12, 8, 10, 9 and 6; now cell 2 has 5 machines 9, 4, 6, 8 and 15 and cell 3 also has 5 machines 7, 9, 12, 10 and 6.

Now, if we each one has 5 machines as described and the product flow will be like this. Now when we do rabbit chasing we actually do not need all these numbers, we just need 2 numbers which is the total which is 45 and the maximum which is 12. Here, the total is 42 and the maximum is 15 and here the total is 44 and the maximum is 12.

Now, what happens is, if we assign now from this we know that, if we have 1 operator then the output is 45. If we have 2 operators it is 22.5. If we have 3 operators it is 15, 4 operators it is 12 and so on. So, here again if we have one operator it is 42, 2 operators it is 21, 3 operators it is 15 and so on. 15 comes because 42 by 3 is 14; 14 is smaller than 15. So, this will be 15.

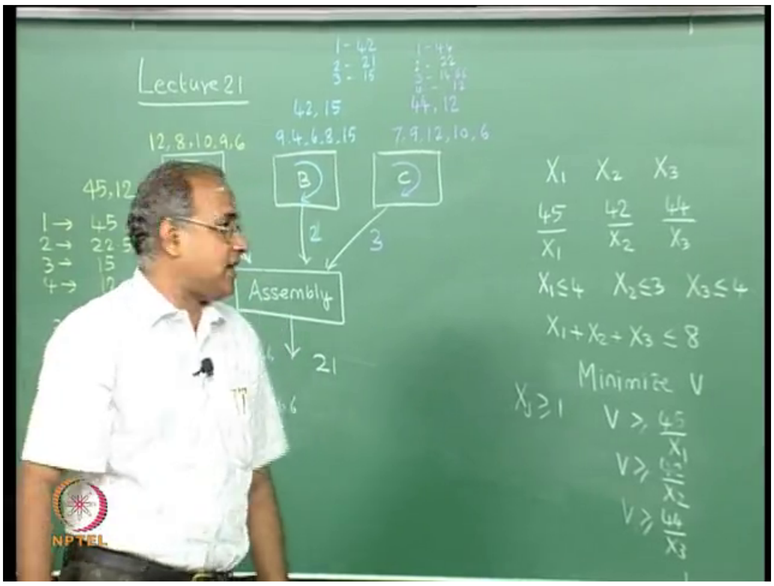
Here, with one operator it is 44, 2 operators it is 22, 3 operators is 14.66 and 4 operators it is 12. So, if we have assign in this case if we assign 2 operators to this and if we assign 3 operators to this and if we assign 2 operators to this. So, if we do 2 comma 3 comma 2 this will be producing a piece at the rate of 22.5 time units, this will be producing a piece at the rate of 15 time units, this will be producing a piece at the rate of 22 time units and the steady state output from the assembly is the maximum of these 3 numbers which is 22.5 time units.

If we do 3, 2 and 3, then by assigning 3 operators to the first one we would get 15, by assigning 2 operators to this one we would get 21, by assigning again 2 operators to this one we would get 22 and the best one will be 22. If we do 2, 2, 3 it would be 22.5, for 2 it will be 21 and for 3 it will be 14.66 and the best value will be 22.5.

So, if we assume that we have 7 operators with us and if we are trying to allocate these 7 operators to these 3 cells, in 3 different ways which is 2, 3, 2; 3, 2, 2 and 2, 2 3 now we observe that the output we get this 22.5, 22 and 22.5. So, the problem is how do we allocate a given number of operators to these 3 cells such that we maximize the productivity or minimize the maximum of the times that we take. The maximum of the times is the rate at which it is coming we want it to be as small as possible.

So, this problem is called the operator allocation problem when we have multiple cells feeding to assembly.

(Refer Slide Time: 22:59)



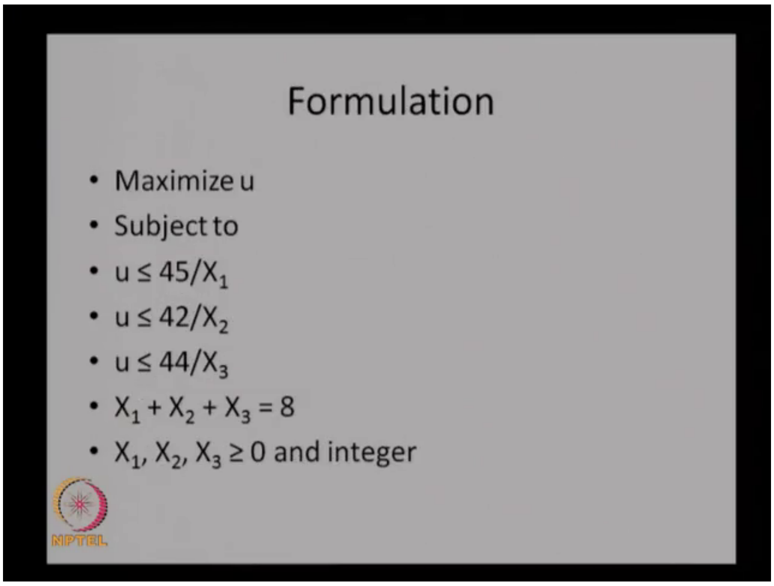
Now, if we assume rabbit chasing, so let us assume that X 1 is the number of operators allocated to the first cell, X 2 is the number of operators allocated to the second cell and

X 3 is the number of people allocated to the third cell. So, when we have X 1 people coming in, the time taken is 45 by X 1, the time taken is 42 by X 2 and the time taken is 44 by X 3.

We also know, that it is not always 45 by X 1 it is 45 by X 1 up to this and this. So, right now, we say that the number of operators that I wish to assign to the first cell cannot exceed 4, because there is even if I increase it beyond 4, I am not going to have any improvement in this. Because, by the very we saw in a previous lecture that even if we have 5 operators here the bottleneck time or the maximum time will take over and this will only be 12 and it will not be more than 12.

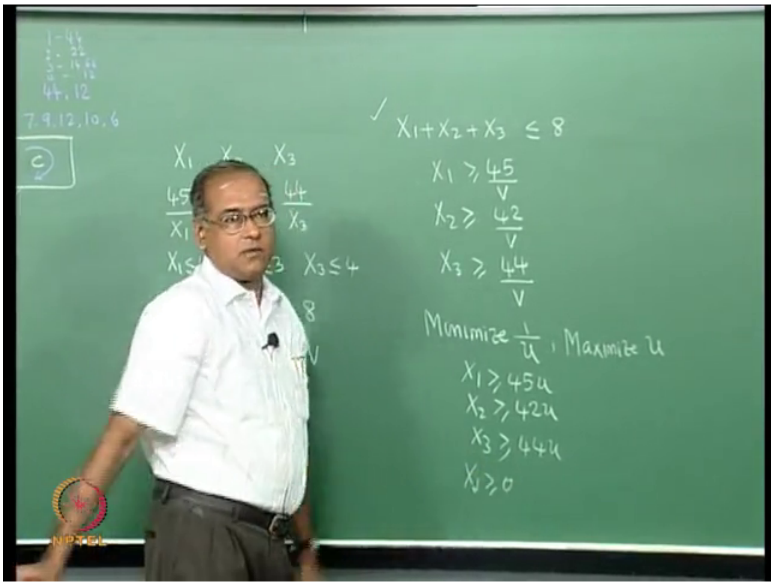
So, there is a restriction that X 1 is less than or equal to 4, X 2 is less than or equal to 3 and X 3 is less than or equal to 4. So, we have these restrictions which are there. Now let us assume that we have 8 operators therefore, we have a constraint in addition to this X 1 plus X 2 plus X 3 is less than or equal to 8. Now, what do we want to do? So, if we assign X 1 operators to the first one, we are going to assume that the output is 45 by X 1, 42 by X 2, 44 by X 3 and therefore, we want to minimize the maximum of these 3 values.

(Refer Slide Time: 25:43)



So, we will say minimize u, we will minimize the maximum of the 3 values, minimize u it's not maximize u. So, let u be the maximum of the 3 values. Now, we want to minimize the maximum of these 3 values. So, we minimize V and define V as the maximum of the 3 values. So, since V is the maximum of these 3 values V is greater than or equal to 45 by X 1, V is greater than or equal to 42 by X 2 and V is greater than or equal to 44 by X 3.

(Refer Slide Time: 26:52)

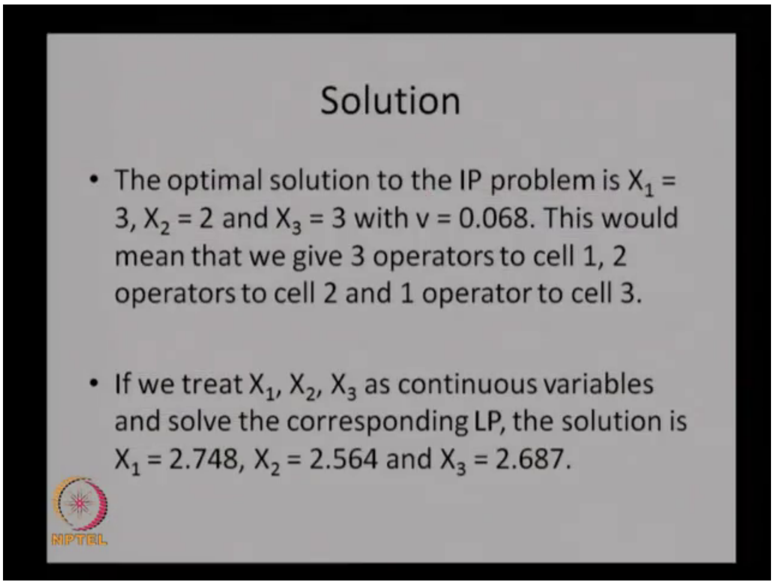


So, the problem now would be to minimize this. So, we rewrite this first and then rewrite it as now we have X 1 plus X 2 plus X 3 less than or equal to 8, because we have a maximum of 8 operators. Now we at the moment ignore these 3, we will use them when required. So, this constraint is written. Now these things have to be written differently. So, this will become X 1 is greater than or equal to 45 by V, X 2 is greater than or equal to 42 by V and X 3 is greater than or equal to 44 by V and we minimize V.

Now, this is still not a linear programming problem because V is a variable X 1 is also a variable. So, this is not a linear programming problem. So, we will now define u is equal to 1 by V and then we say that minimize 1 by u is equal to 1 by V. So, minimize 1 by u subject to X 1 greater than or equal to 45 u, X 2 greater than or equal to 42 u and X 3 greater than or equal to 44 u plus the constraint X 1 plus X 2 plus X 3 less than or equal to 8.

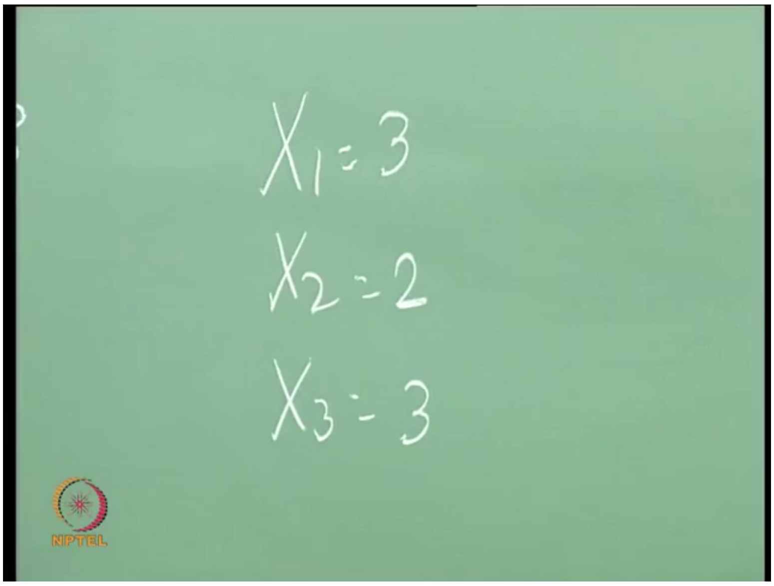
Now, minimizing 1 by u is the same as maximizing u. So, now, this becomes a linear programming problem that maximizes u subject to X 1 greater than 45 u, X 2 greater than equal to 42 u, X 3 greater than or equal to 44 u, X 1 plus X 2 plus X 3 less than or equal to 8 and X j greater than or equal to 0.

(Refer Slide Time: 29:09)



So, this becomes a linear programming problem and when we solve this problem we get X 1 equal to 3, X 2 equal to 2 we get a solution.

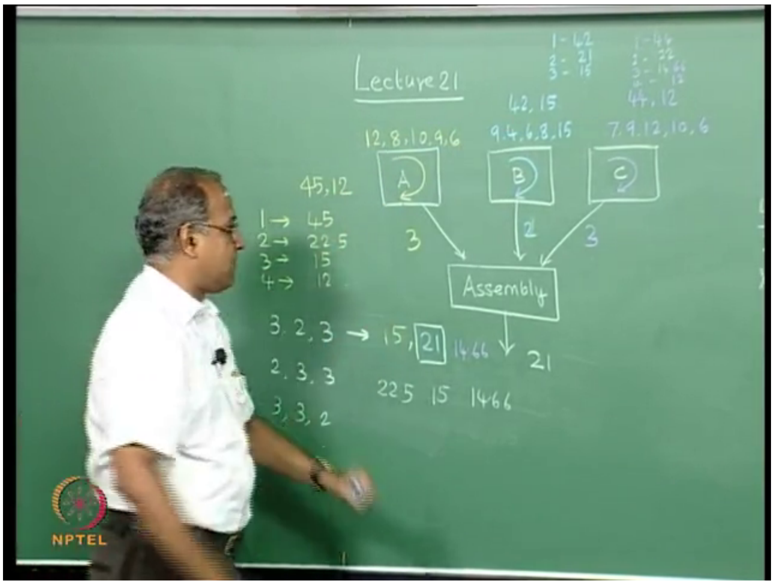
(Refer Slide Time: 29:22)



X 1 equal to 3, X 2 equal to 2 and X 3 equal to 3, we get a solution this way.

Now, this would mean that with 8 operators we allocate them in the order 3 comma 2 comma 3.

(Refer Slide Time: 29:42)



So, when we have 3 this person would be producing at 15, when we have 2 this one would be producing at 21 and then we have 3 operators the third cell will be producing at 14.66 and the maximum of that will be 21 which will be the steady state output of the assembly.

Now, when we look at 21 as the steady state output of the assembly, then we also have to look at one more aspect here. So, we are assign 3 operators here in the end we have 2 operators here for 8 and then we have 3 operators here for the third cell. Now, what happens is, if we allow this many operators to work this person is going to produce at the rate of 15 minutes per piece, 21 minutes per piece and 14.6 minutes per piece. Steady state output from this system is 21 minutes per piece or 21 time units per piece.

Now, if we allow these numbers of operators to work continuously then what happens is, this is the bottleneck. This is the cell that is creating the 21 time units. So, these 2 cells will over produce. So, either we have to make sure that they do not produce more or we will end up building inventory in these 2 places. If we do not want to build inventory in these 2 places then, we have to control the amount of material that comes into this and comes into this. As previously mentioned they lead to Just In Time manufacturing models where we control the amount of material that come into these 2. The net effect of it would be that the operators here would be underutilized and will not be utilized fully.

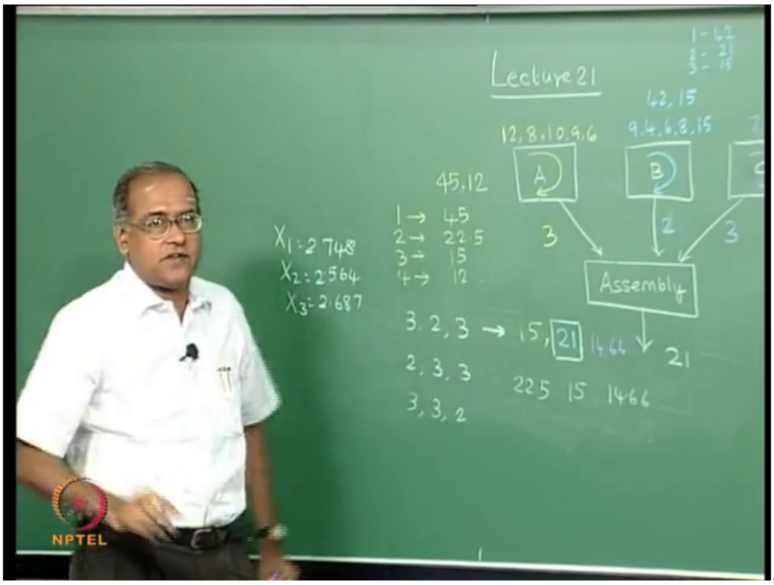
So, once again we will get back to shifting operators so that, the person the roster will shift operators from one shift to another, so that the same person does not get overworked. This again leads us to the fact that all the operators will have to be cross trained and be capable of working on all the machines in this context. Another possibility also is that, if building a little bit of inventory is permitted then what we can do is, let them work at their full speed, build some inventory here and then after some time shift an operator from here to this or from here to this; because, this cell can have a maximum of 3, beyond which it is not producing more than the rate of 15.

So, after some inventory is built up one person can move and work here. So, let this 21 becomes 15. This will slow down because this from 15 will become 22.5. So, after some time we can adjust it and make it as 2, 3, 3 with output 22.5, 15 and 14.66. Even though this is producing at 22.5 there is some inventory that is built up here which can be consumed and now the steady state output will come to 15.

Then after some time this inventory will come to 0. So, we will shift this operator back, by which time some inventory could be built up here and so, shift one person here. So, we will then move to 3, 3 and 2. 2 would mean that this is this person will produce only 22 will take 22 time units per piece, but there is some inventory built up here.

So, with 3 and 2 will be operating at 15, 15 and 22, but that 22 is offset by the fact that there is some inventory and so on. So, if we keep doing this then typically we are solving essentially like a linear programming problem.

(Refer Slide Time: 34:46)



So, the solution is 2.748, X 1 is equal to 2.748, X 2 is equal to 2.564 and X 3 is equal to 2.687. This is like on an average we are working at this way. The 3 and 3 at some point the operator will move from here to here which means the 2 will increase, this will come down. At some point one operator moves from this to this, so, this will work at the steady state 2.564 while the other thing will work at 2.748, 2.687. Now, these numbers come because 45 by this, will be equal to 42 by this, will be equal to 44 by this.

So, at steady state the outputs from all the 3 cells will be equal; provided, we allow a small amount of inventory to be built up here and here and then shift these operators. So, it becomes what is called a dynamic way of reallocating operators, building a some amount of inventory, so that, the operators are not underutilized and most importantly by doing s o, we are able to increase this output. From 21 it will move to a balance which is equal to the same thing which will be 45 by 2.748, 42 by 2.564 and 44 by 2.687. So, we will move to that number which is less than 21 time units per piece.

Now, another way of looking at this problem is when we actually solved this there are 2 other things that we will have to look at; one is, in a way we are making a certain approximation because this 45 by X 1, 42 by X 2, 44 by X 3 are not exactly the same because the output is not 45 by 4, if we put 4 values for X 1 the output is not 44 it is not 11.25, but it is 12.

So, 45 by X 1 will hold only when X 1 is between 1 and 3. When X 1 if X 1 is 0 then this will start behaving in a very funny manner, but the very problem the very nature of the allocation problem will ensure that we do not allocate 0 to any cell. So, we could put another condition here such that X j is greater than or equal to 1. So, that we do not allocate 0, because if we allocate 0, 45 by X 1 will go to infinity. As far as the linear programming problem is concerned if X 1 is equal to 4 this would take the steady state output as 45 by 4 and will not take it as 12.

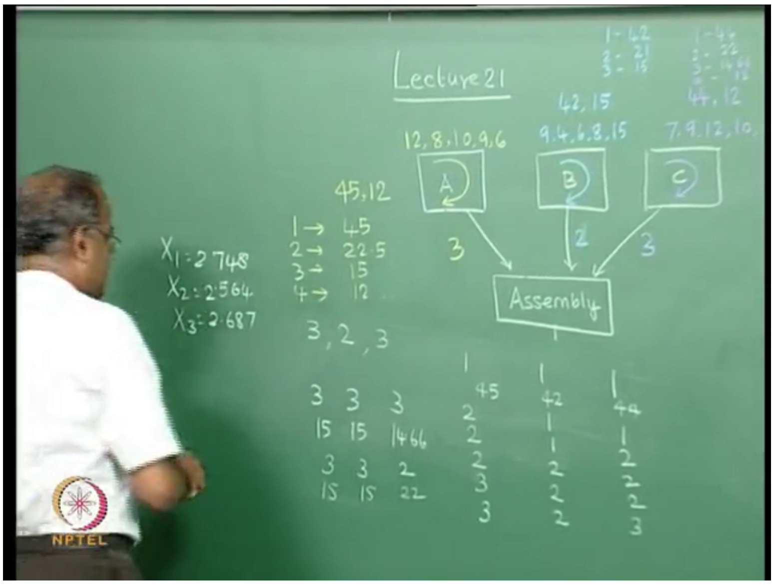
So, we have to be conscious we have to be aware of this fact and so, we first solve this problem once, then see what kind of an answer we get and then progressively do it by adding these types of constraints and then by adjusting the values at some point. So, there is a kind of a iterative approach, there is a kind of a redoing this problem depending on what kind of a solution we get. One way of overcoming that is by taking the linear programming solution and by working with it.

So, if we take the linear programming solution, the linear programming solution is 2.748, 2.564 and 2.687 with 8 operators. So, first thing what we do is, you allocate one operator to each cell. So, when you have one operator to each cell, in fact, from the linear programming solution we could do this. What we could do is the upper, the closest integer value for this is 3, the closest integer value for this is 3, the closest integer value for this is also 3. So, we cannot have 3, 3 and 3.

At the same time we will have to find out which variable has the largest fractional value. So, this one has the largest fractional value. So, 2.7 is approximated to 3. The next one that has largest fractional value is this. So, this is approximated to 3 and since there is a total of 8, this will have to take a value 2 which was our optimum solution with 3, 2 and 3.

So, many times the LP optimum suitably rounded, I say suitably because if we actually round it we will get 3, 3 and 3. 2.564 will be rounded to 3, but then we do not have 9 operators. So, we will suitably round them based on the largest fractional component. The variable that has the largest fraction first becomes takes it is upper integer value and this is progressively done till we utilize all the operators that we have.

(Refer Slide Time: 40:49)



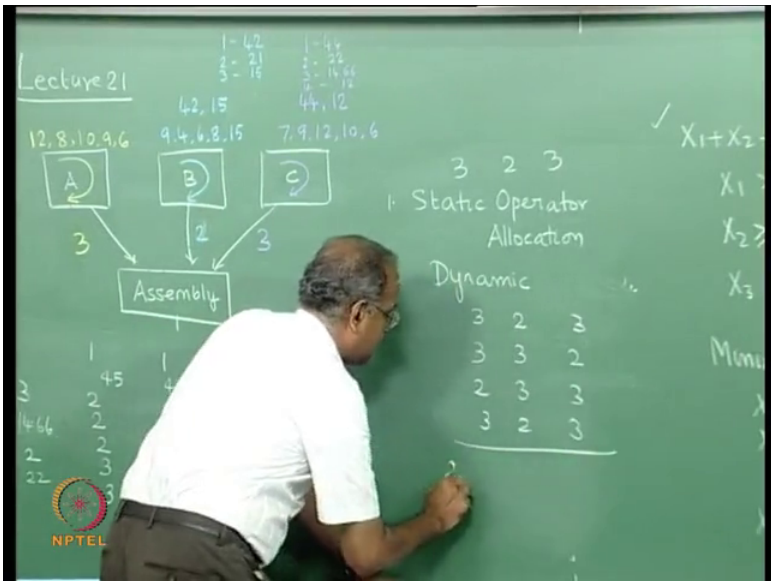
So, that is one way to do this. The other way to do this is to start with one operator for each cell and if we do that our output will be 45, 42 and 44. Now, this is the slowest, give one more person to the slowest. So, you get 2, 1, 1. Now, this becomes 22.5, this becomes 21, this becomes 42, this becomes 44, this is the slowest one. So, you give 2, 1, 2. So, this is 22.5, this is 42, this is 22, this is the slowest, you get 2, 2 and 2. Now, this is 22.5, this is 21, this is 22, this is the slowest, give one more person 3, 2, 2. Now, this becomes 15 this becomes 21, this becomes 22. So, this is the slowest, give one more person you get 3, 2, 3, my 8 people are exhausted therefore, I get this as a solution.

So, this heuristic solution almost in all cases gives the optimum solution another way to do is this is my LP optimum. So, put everything to its corresponding upper integer value. So, I would get 3, 3 and 3 with 9, whereas, I have only 8 operators. So, with 3, 3 and 3 my output will be 15 here, my output will be 15 here and my output there is 14.66 here.

Now, you have to reduce one. So, reduce from the fastest. So, if you do that, if you reduce from the fastest you will get, this is the fastest 14.66 is the fastest. So, reduce from the fastest you get 3, 3 and 2, which would give you 15, 15 and 22. So, you do not get the optimum solution, you get 22 as a steady state output for this heuristic solution. Increase everything to its upper value and then take away till you get the number of operators that you have. So, you could do this one also.

So, these are the several ways by which we look at what is called operator allocation problems to manufacturing cells.

(Refer Slide Time: 44:05)



What we have actually done is, we have actually done 2 aspects of this problem one is what we call as a static operator allocation and we have looked at what is called dynamic operator allocation.

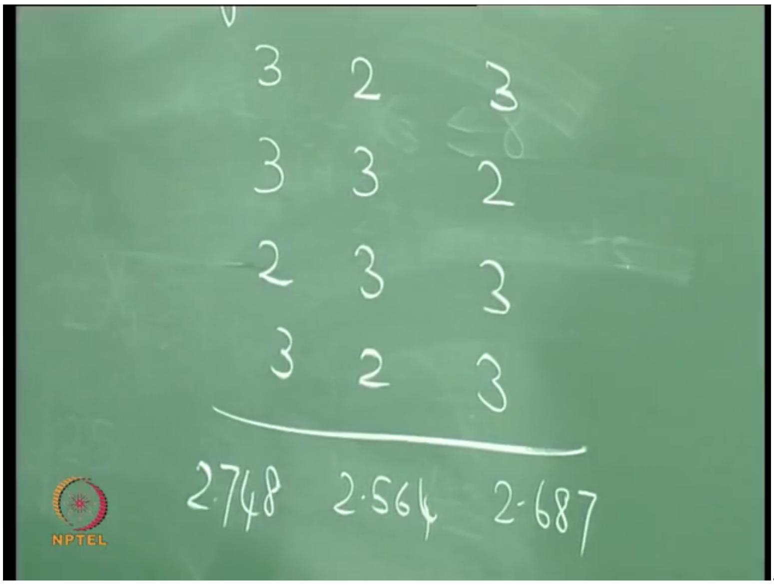
So, in the static operator allocation case where we could implement this solution which is 3, 2 and 3, which would give us an output of 21 time units. But, as I said this will be coming out at one piece per 15 time units, one piece per 21 time units, one piece from 14.66 time units.

So, once again, in the static case either there will be inventory or there will be under utilization if we control the number here. If we control the inventory here and let this produce at the rate of 21, this will be producing at the rate of 21 time units per piece. We control inventory and make this produce at the rate of 21 per piece. The steady state output will be 21. Now, in order to ensure do rotation, so that, the roster will ensure that no operator is unfairly over utilized.

The other way, as I said is to try and build the inventory and then move operators from one cell to another, dynamically. So, we would start with 3, 2, 3, static would be 3, 2, 3, dynamic we would start with 3, 2, 3 and then after a while we build up inventory, now out of the 3 this is the fastest. This for 3 people is 15, 14.66 and 21. So, this is the fastest. This is also building inventory, this is also building inventory. So, then one person will come from here to this. So, we will operate at 3, 3, 2 for some time and then we realize that this is getting a little faster both are depleting at a faster rate because with 3 people here 21 also starts coming to 15.

So, all the inventories that are built up is getting little depleted this is producing at 15, this is producing at 15, this is producing at 22, but it is getting depleted very quickly. So, then after a while we would do 2, 3. 3 and then we will come back to 3, 2, 3 we will do this.

(Refer Slide Time: 47:13)



Which means at steady state we will be it is equivalent of 2.748, 2.564 and 2.687.

Now, in this model we have assumed that there is only one product and all these 3 manufacturing cells will feed to the assembly. So, a very logical extension would be when we have multiple products. Now, when each cell makes let us say the same product, but 2 variants of the same product. The 2 variants again may require 3 parts or 3 sub assemblies. Each is made in a cell and then it is brought into the assembly then we also have to look at operator allocation in the context where we have multiple cells making multiple products. The model is very similar to what we have seen now with a very slight difference and we will look at that model in the next lecture.