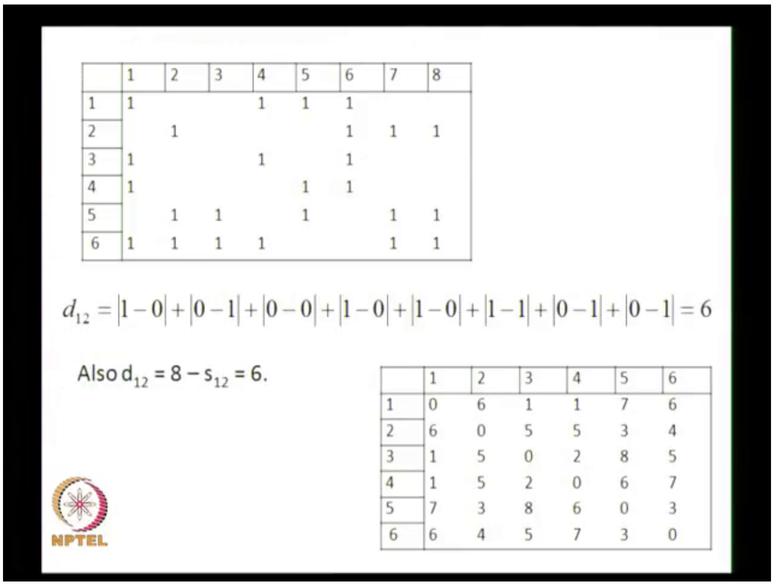
Manufacturing Systems Management
Prof. G. Srinivasan
Department of Management Studies
Indian Institute of Technology Madras

Lecture - 13
Assignment model, Algorithm considering sequence of visit of machines

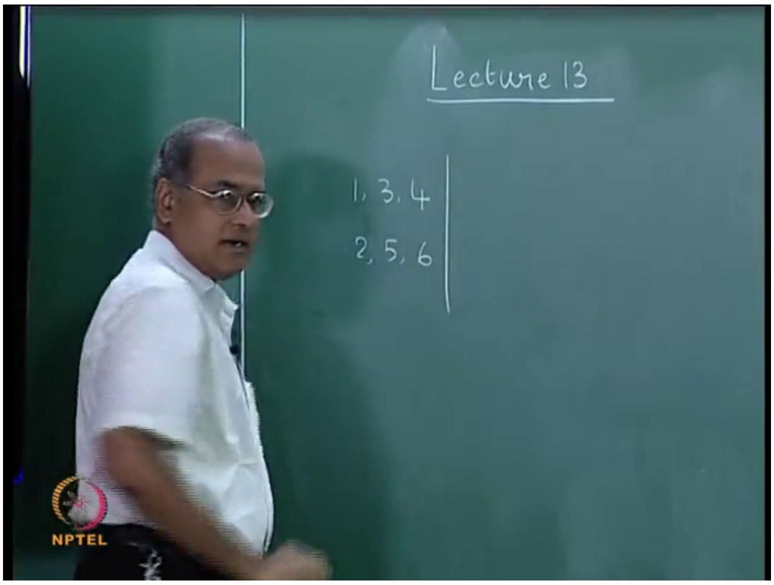
In this lecture, we continue the discussion on the assignment model, we ended the previous lecture by saying that we could encounter situations, where the number of groups obtained for machines could be more than that for the parts and the also said that the algorithm has to be suitably modified to take care of such a situation.

(Refer Slide Time: 00:53)



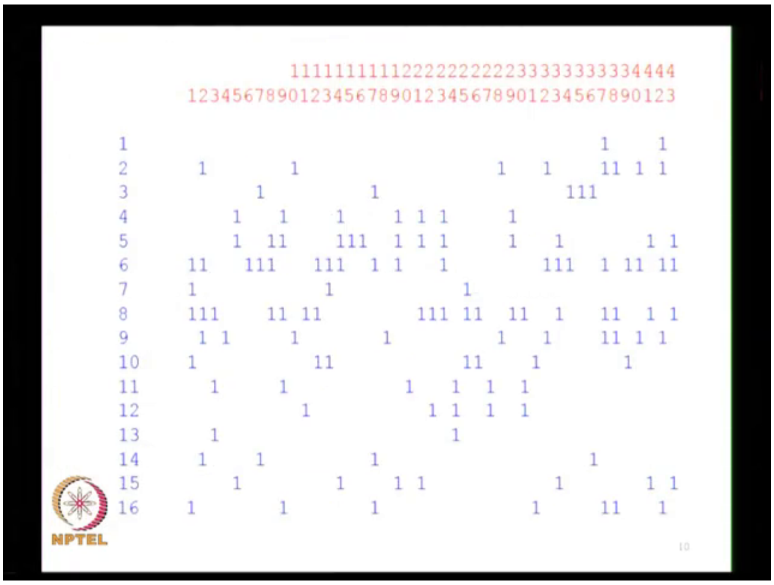
Now, this situation would happen particularly when we are solving a slightly larger sized problem. The 6 machine 8 part problem that we have used to explain the algorithm is a small sized problem. So, the assignment solution gave 2 groups and both the groups is able to attract part families.

(Refer Slide Time: 01:11)



For example for this the machine groups are 1, 3, 4 and 2, 5, 6 and each of these was able to attract part families. Now let us explain the algorithm using a large sized matrix.

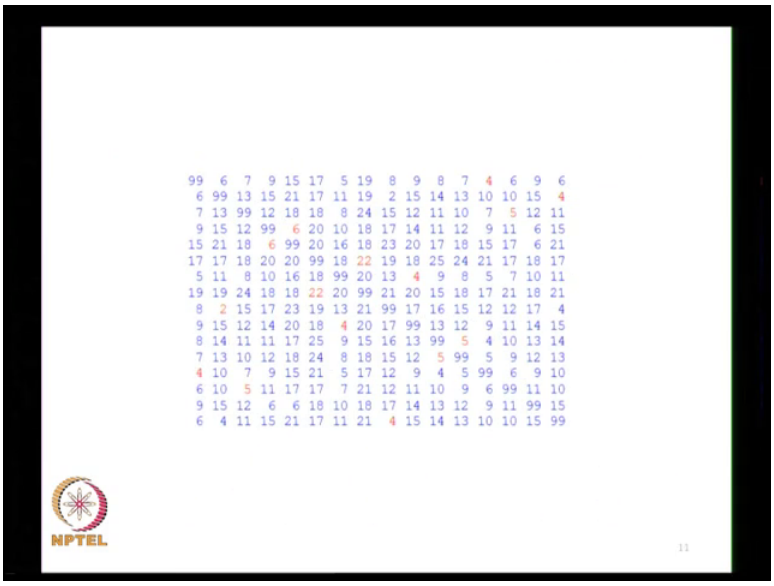
(Refer Slide Time: 01:25)



This matrix we have used when we worked out the production flow analysis solution, this is a same 16 machine 43 part matrix used by king Nakornchai earlier by and so on.

Now, let us apply the assignment based algorithm to solve this to understand a couple of issues related to the algorithm. So, we first create a distance matrix for machines a dissimilarity and distance are the same, so we create a distance matrix for machines and that is a 16 by 16 matrix. So, I am not going to explain the entire computation. But you can see this distance by this distance matrix which is a 16 by 16 matrix.

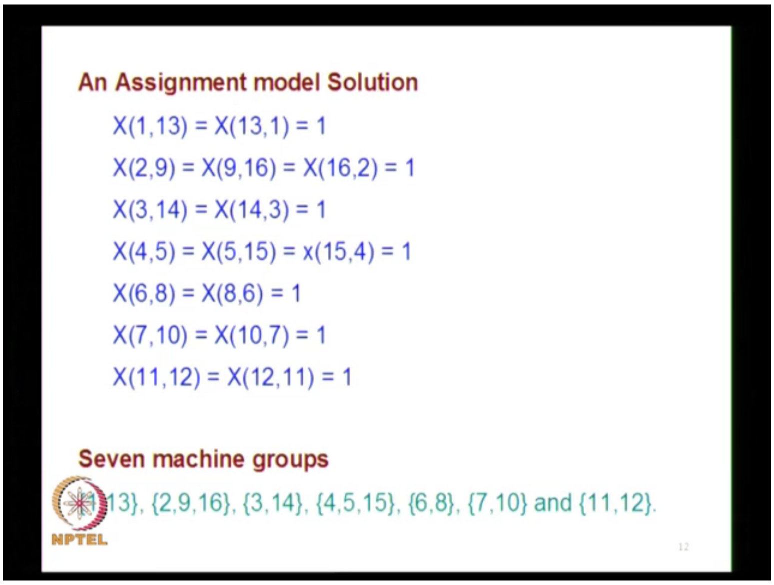
(Refer Slide Time: 02:18)



99	6	7	9	15	17	5	19	8	9	8	7	4	6	9	6
6	99	13	15	21	17	11	19	2	15	14	13	10	10	15	4
7	13	99	12	18	18	8	24	15	12	11	10	7	5	12	11
9	15	12	99	6	20	10	18	17	14	11	12	9	11	6	15
15	21	18	6	99	20	16	18	23	20	17	18	15	17	6	21
17	17	18	20	20	99	18	22	19	18	25	24	21	17	18	17
5	11	8	10	16	18	99	20	13	4	9	8	5	7	10	11
19	19	24	18	18	22	20	99	21	20	15	18	17	21	18	21
8	2	15	17	23	19	13	21	99	17	16	15	12	12	17	4
9	15	12	14	20	18	4	20	17	99	13	12	9	11	14	15
8	14	11	11	17	25	9	15	16	13	99	5	4	10	13	14
7	13	10	12	18	24	8	18	15	12	5	99	5	9	12	13
4	10	7	9	15	21	5	17	12	9	4	5	99	6	9	10
6	10	5	11	17	17	7	21	12	11	10	9	6	99	11	10
9	15	12	6	6	18	10	18	17	14	13	12	9	11	99	15
6	4	11	15	21	17	11	21	4	15	14	13	10	10	15	99

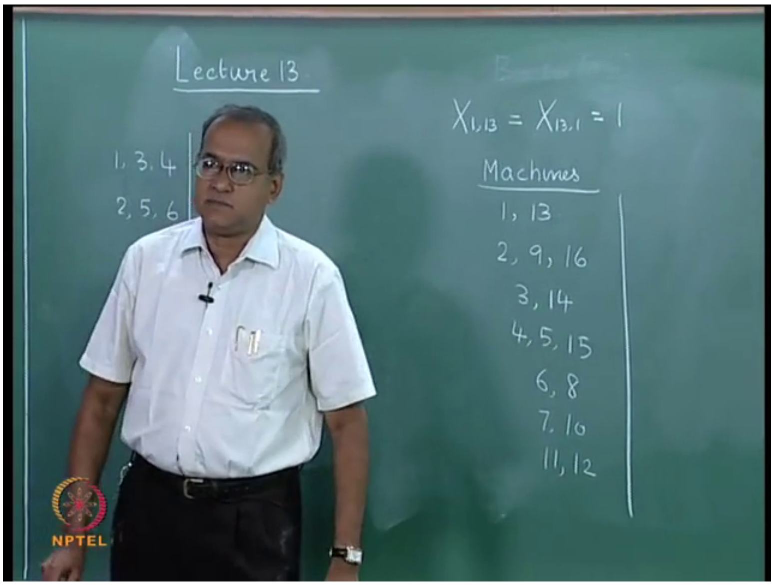
There are 16 rows and columns and this will be used as an input to solve the assignment problem, this is a distance or dissimilarity matrix. As mentioned in the previous lecture they do not want diagonal assignments because, diagonal assignment would mean assigning a machine to itself which in a way would represent a manufacturing cell with only 1 machine, since we do not want diagonal assignments we give a value infinity to the diagonal elements and for operational purposes we give a large value to the diagonal element, so that diagonal assignments do not happen. So, in this matrix you will observe that the diagonals take a value 99 which is significantly higher than any other of diagonal number. So, 99 would serve the purpose of a large number and will prevent diagonal assignments from happen.

(Refer Slide Time: 03:29)



Now, let us solve an assignment problem out of this and the assignment solution that we have is this with 1 13 X 1 13 equal to X 13 1 equal to 1.

(Refer Slide Time: 03:44)



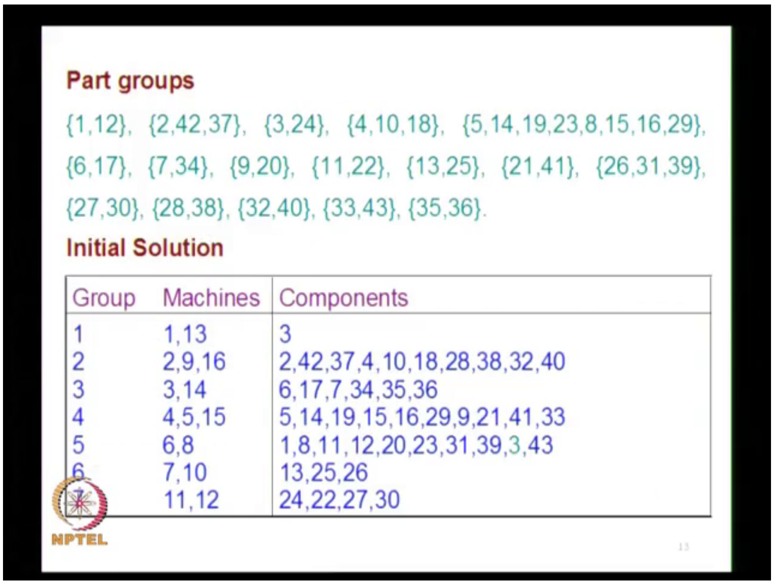
We look at the slide we realize that there are actually 16 assignments, 1 to 13, 2 to 9, 3 to 14, 4 to 5, 5 to 15, 6 to 8, 8 to 6, 7 to 10, 8 to 6, 9 to 16, 10 to 7, 11 to 12, 12 to 11, 13 to 1, 14 to 3, 15 to 4 and 16 to 2. So, we see this carefully 1 is assigned to 13 and 13 is assign to 1, which creates a machine group of 1 and 13 belongs to 1 group.

2 goes to 9, 9 goes to 16, 16 goes to 2, so we will have a solution 2, 9, 16, 3 goes to 14 and 14 goes to 3, So 3 to 14, 4 goes to 5, 5 is assign to 15, 15 is assigned to 4. So, machines 4 5 and 15 will form another group, 6 goes to 8 8 goes to 6 so 6 and 8 7 10 11 and 12.

So, the assignment solution would give us 7 groups which are also shown on the lower portion, now we also observe that this matrix is symmetric and because of the symmetry of this matrix there will be a tendency to have something like X 1 13 equal to 1, X 13 1 will be equal to 1, similarly 2 to 9. 9 to 16. 16 to 2. So, symmetry would take us closer to these groups with size smaller size 2 or 3 or whatever, this actually helps us in our cell formation algorithm because this would give if we start with m in this case 16, we would get we have prevent a diagonal assignments. Therefore, the number of groups that we get should be m by 2 or less because by preventing a diagonal assignment we have made sure that each 1 of these has a size of 2 or more, therefore we will have m by 2 or less.

If each one of them has a size 2 then we would have got 8 groups here for 16 machines, now 2 of them have size 3 so we have 7, but the number of groups that we will get here will be close to m by 2 and we realize that they something that is desirable at this stage. So, now what we do is we start with this groups, these groups are shown here.

(Refer Slide Time: 07:04)

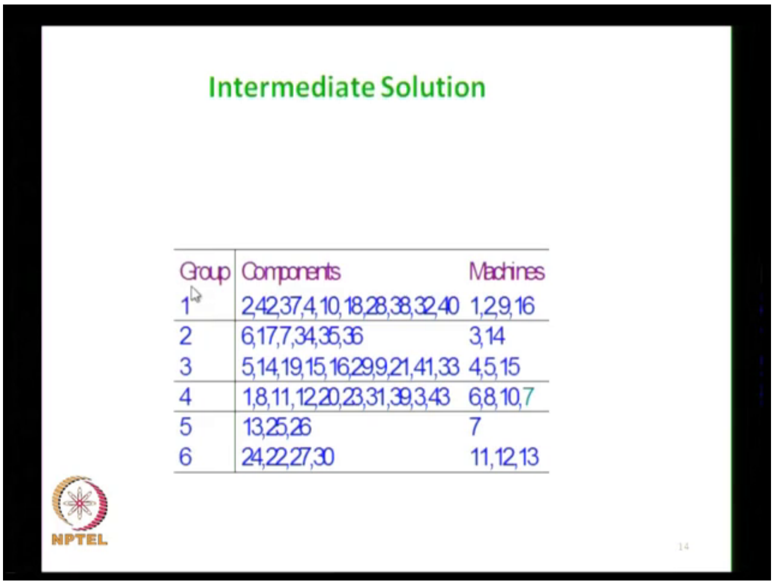


Shown here the 7 groups are shown here and then we apply the part assignment rule to assign these 43 parts to the 7 groups, now when we do that we observe that now we go back to the this matrix and then start looking at each column and then try to find out what are all the machines each part requires. For example part 1 requires 6 7 8 10 and 16, so, part 1 will come here 6 8 7 10 and 16. So, here it visits 1 here it visits 2 here it visits 2. So, based on our part assignment rule it should go to the machine group where it visits maximum number of machines.

So, it can go to either 6 8 or it can go to 7 10 there is a tie. now then tie is broken by the smallest size both of them have size equal to 2, so now, it is assigned to 6 and 8 it could have been assigned to 7 and 10 also, like this we assign all the 43 components or parts to these 7 groups. Now when we assign these 43 parts to the 7 groups, we realize that this group 1 13 that has machines 1 and 13 has got only 1 components which is component number 3, this has several components all other groups have a reasonable number of components, where as this group is able to attract only 1 component.

So, then we ask a question do we want a cell to have only 1 machine or 1 part or component or should we restrict every machine group or a part family to have at least 2 members. So, if we restrict that every group machine group and part family should have at least 2 members, then this 3 should not be assigned here. So, 3 goes to the next best group and 3 goes to this. Now when we reassign this 3 to this we realize that we have started with 7 machine groups, but there are only 6 part groups, so there is an unequal number of machine groups and part groups. So we do not treat this as a solution, we treat a solution as acceptable when we have an equal number of machine groups and part groups now we start with 7 machine groups, now we get 6 part groups; now for the 6 part groups are component groups we again compute the machines by using the machine assignment rule.

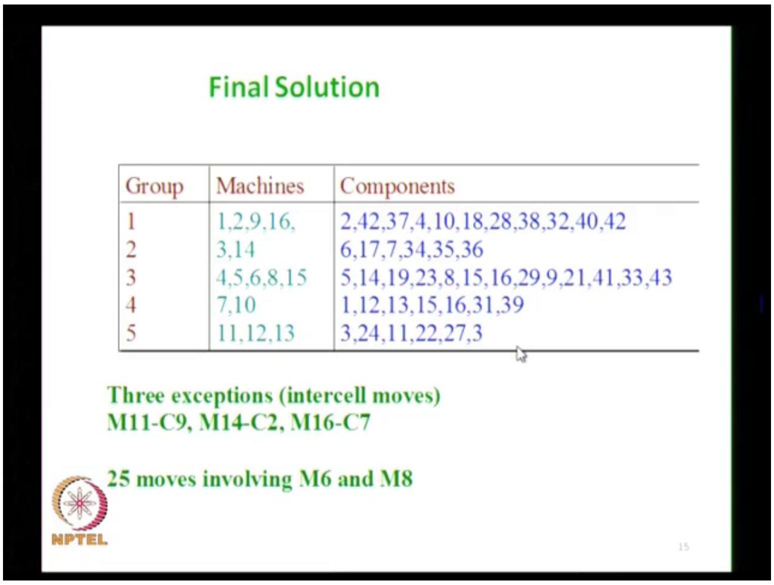
(Refer Slide Time: 10:35)



So, we now keep these as base which is shown in the next slide, now the next slide has only 6 groups with this as base and now we start reassigning the machines, for example we go back to machine number 1 here component 37 and 42 are the components. So, look at this 37 and 42 are here so machine 1 goes here, similarly we assign all the machines. Now when we remember that 6 and 8 also will go to only 1 group and will not go to multiple groups, in production flow analysis at the end for the same matrix we said that we could duplicate 6 and 8 right now we do not do that, when we assign machine 6 and 8 we say that they will go to only one group.

Now when we do this we realize that this part group number 5 has attracted only one machine which is machine number 7. So, we once again go back and ask that question do we have a machine group with only 1 machine or do we have put a restriction that we should have at least 2 members. So, under the condition that we should have at least 2 members, we now reassign machine 7 to it is next best group and machine 7 goes here. So, now we have 6 component families to begin with and the machine assignment rule has given us only 5 machine cells. So, we do not treat this as a solution and we proceed again by applying the part assignment rule to these 5 machine groups which is shown in the next slide.

Refer Slide Time: 12:24)

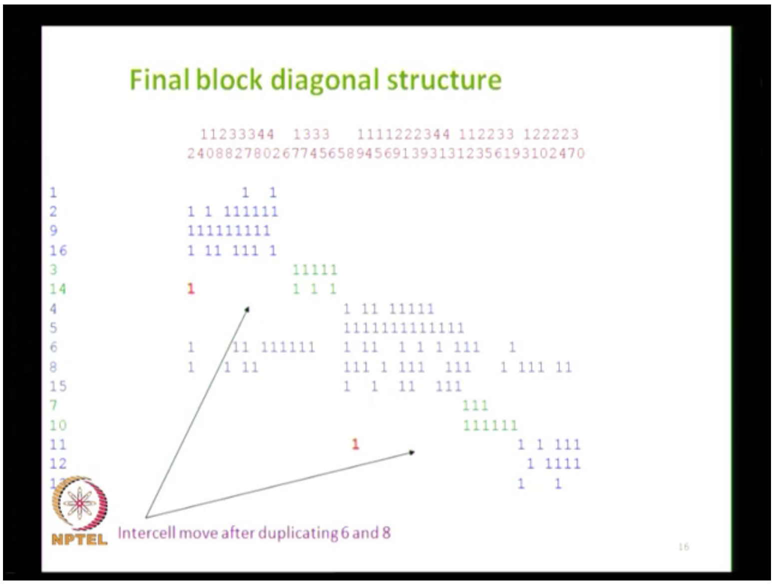


Now, these are the 5 machine groups to which these parts are assigned, now when we do this is the part assignment once again we could go back to the matrix look at every column and then assign the part to the correct machine group, using the part assignment rule and when we do that we have this solution. Now we have an equal number of machine groups and part groups, now we have 25 moves we have calculate the intercell moves, here now the algorithm continues so that can there be an improvement.

Now with this part groups or component groups apply the machine assignment allocation rule and try to get machine groups, when we do this we get the same set of machines repeating here and therefore, the algorithm will terminate. So, what will also happens is if you go back to the starting solution for 16 machines; we started with 7 groups and when we apply this condition that a machine cell or a part family should have at least 2 members the number of groups automatically reduces to a very desirable nice answer in the end.

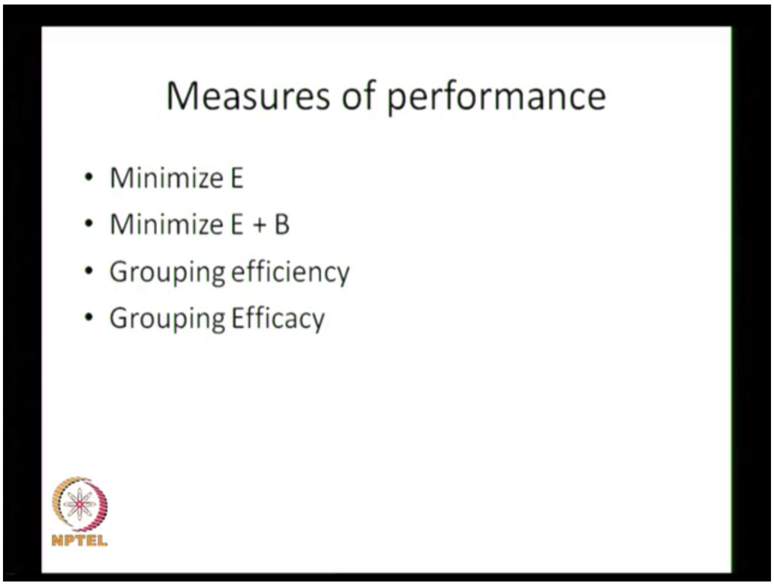
Now, if we go back to pfa solution, we realize that this is a solution that we got using production flow analysis.

(Refer Slide Time: 14:00)



So, this is the final block diagonal structure, with 2 intercell moves that we had seen in the pfa provided 6 and 8 are sufficiently duplicated right, now 6 and 8 are not duplicated in this solution because 6 and 8 appear here 4, 5, 6, 8, 15, so 4, 5, 6, 8, 15, 6 and 8 are here. So, with a large number of intercell moves, but since these machine groups repeat the algorithm terminates by giving this as the final solution.

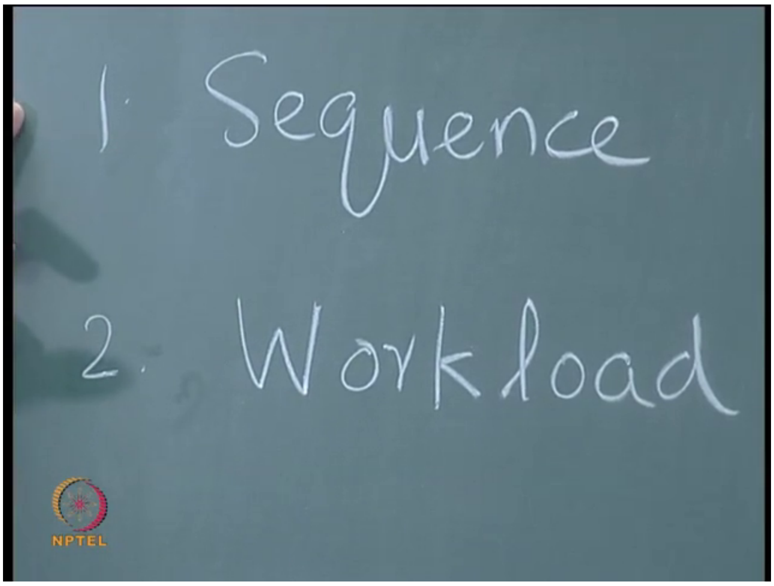
(Refer Slide Time: 14:39)



So, this is how the assignment based algorithm can be used for cell formation, now we here almost going to close the discussion on cell formation algorithms considering only 0 1 binary or incidence data.

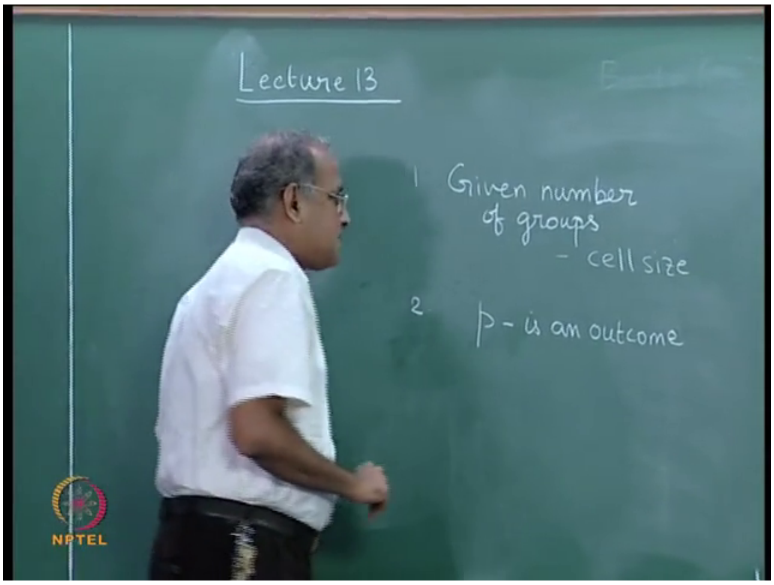
If we go back to earlier lectures, we have said that the binary form of representing the incidence does not take into account. The operating sequence and does not take into account work load.

(Refer Slide Time: 15:22)



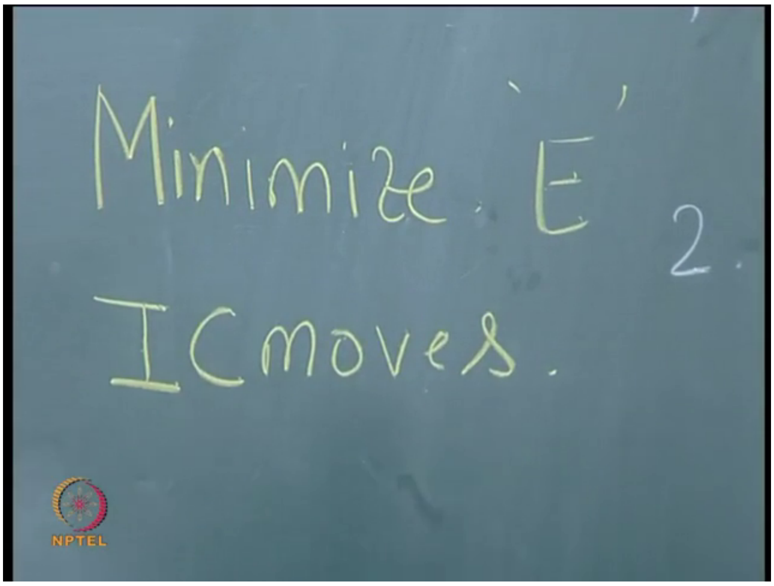
So, we will be looking at 1 or 2 algorithms that address this aspect of the problem, but before we do that let us also spend some time to try and understand some measures of performance, now what are these measures of performance? If you go back and look at the algorithms that we have seen, some of the algorithms we will assume the number of groups as an input, particularly all the optimization based algorithms expected the number of groups to be an input to the algorithm, whereas most of the heuristic algorithms that we saw including production flow analysis, rank order clustering, assignment based algorithms the similarity base algorithms.

(Refer Slide Time: 16:39)



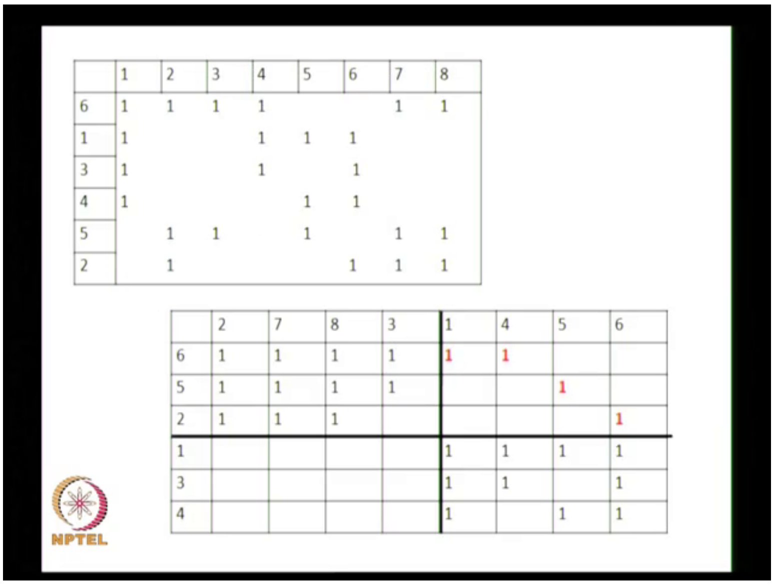
A number of groups was an outcome and not an input. So, we could think in terms of 2 things given number of groups and 2 numbers of groups as a variable or as an outcome. So, p is an outcome or output of the algorithm, now the moment we have given number of groups. In fact, a optimization algorithm also required a cell size the algorithm required that we restrict the size of the cell to some capital N.

(Refer Slide Time: 17:27)



So, in all these cases minimizing intercell moves is the objective, now E is given as the intercell moves so minimize E which is minimize the exception.

(Refer Slide Time: 17:54).



	1	2	3	4	5	6	7	8
6	1	1	1	1			1	1
1	1			1	1	1		
3	1			1		1		
4	1				1	1		
5		1	1		1		1	1
2		1				1	1	1

	2	7	8	3	1	4	5	6
6	1	1	1	1	1	1		
5	1	1	1	1			1	
2	1	1	1					1
1					1	1	1	1
3					1	1		1
4					1		1	1

For example if you see a final structure like this is the 6 by 8 incidence matrix that we have been using, this is the final solution with machine groups 1, 3, 4, 2, 5, 6 and the corresponding part groups, now these 4 elements shown in red color are the intercell moves, they are intercell moves if we create 2 machine groups and 2 part groups and when the parts move they represent intercell moves.

If the same thing is represented in the form of a binary matrix which is what we have done here, these 4 are called exceptions or exceptional elements, they are expected to lie within the diagonal block these are the 2 diagonal blocks, each diagonal block represents a cell with machines and parts. If all the ones are within the diagonal blocks then there is no intercell move, there are no exceptions now these are called exceptions which are called E.

So, minimizing E is a measure of performance and it is an expected measure of performance, but minimizes exceptions provided the numbers of groups are given. Now the off diagonal ones are called exceptions or intercell moves, there is a slight difference between a exception and intercell move which we will see when we get into operating sequence, but otherwise or unless otherwise stated exceptions and intercell moves can be treated as the same in the context of a binary matrix.

Now, we expect all these ones to be inside the blocks now this is called block diagonal zing the given matrix. So, minimize E is an objective now let us look at this while this 1 represents that part number 4 which is assigned to this group, visits a machine here which is an intercell move. Now what is this blank represents or what does this blank represents? Now this blank or a 0 represents that component number 3 even though it is assigned to this group it does not require one of the machines. So, it represents under utilization in some form not exactly directly under utilizations, but it can result in some kind of an under utilization, particularly in this case because while machine 2 is assign to this there is a part that is there in it is own group which does not want machine 2, but there is some other part belonging to some other group which wants machine 2.

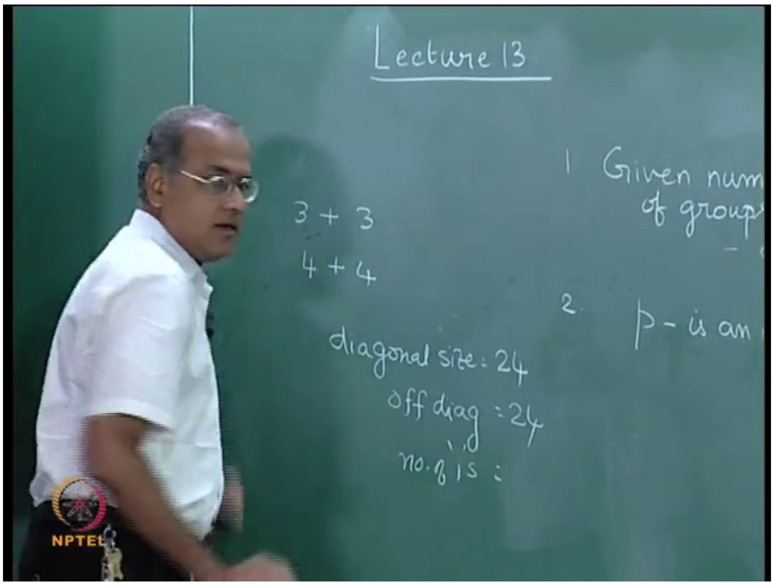
So, this could represents some kind of under utilization by restricting machine 2 to this group and if for some reason we are subcontracting this operation, then we are creating under utilization of machine 2 in 1 group and there is a operational subcontracting on the other to minimize intercell moves. So, the presence of the 0 or a blank within the diagonal block represents some kind of an under utilization and also not very desire, particularly if you look at this problem as a block diagonalization problem the presence of a 0 inside the block is also not desire.

From a practical sense the presence of a 1 outside the block is far more undesirable, than the presence of a 0 inside a block from a practical sense. From a practical sense this 1 is an intercell move, this 0 it actually does not matter from a practical sense if part number 3 is assigned to this group and does not visit machine 2 it is, but from a block diagonalization perspective as a mathematical problem to block diagonalize the 0 is also undesired. So, sometimes we also look at minimizing E plus B, B is the number of blanks inside E is the number of exceptions outside. So, we want to minimize E plus B. So, in this solution if I am minimizing E then the number of exceptions is 4, if I minimize E plus B it is 4 plus 2 4 plus 3 there is 1 here there are 2 here.

So, sometimes minimizing E plus B is also an objective or a measure of performance, but these measures are largely for the block diagonalization problem and not for the cell formation problem, even though the cell formation problem and the block diagonalization problems are similar there is a slight difference because, in practice the presence of a blank is not as serious as the presence of an intercell. Whereas for a block diagonalization problem it is kind of indifferent between a 0 and a 1 and therefore presence of a 0 inside the diagonal is as important or unimportant as presence of a 1 outside the diagonal, so E plus B is another objective. Now there are 2 either objectives given which are called grouping efficiency and grouping efficacy, I will kind of explain both of them very quickly both are measures for block diagonalization, the goodness of the block diagonalization.

Now, if this is the solution there are 2 groups the sizes are 3 plus 3 and 4 plus 4, what I mean by 3 plus 3 is this machine group has size 3 this has size 4.

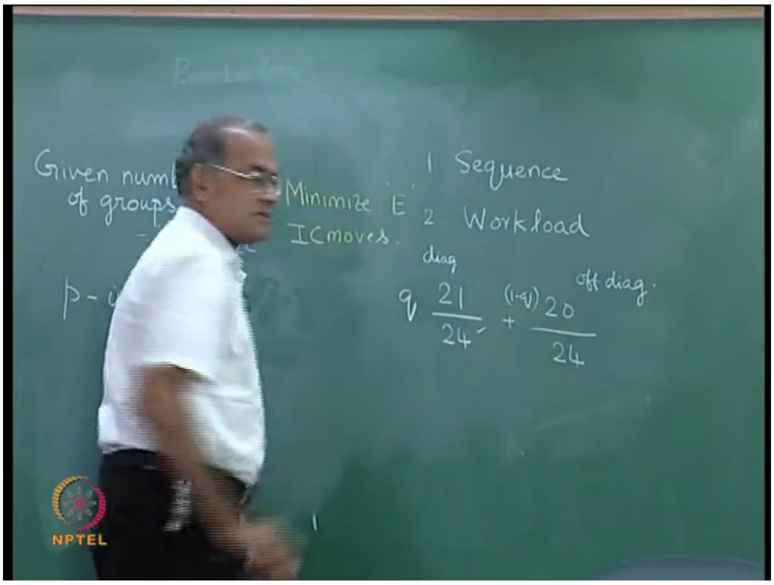
(Refer Slide Time: 24:27)



So, the 2 diagonal blocks have a size of 12 plus 12, this is a size 12 this is another size 12. So, the 2 diagonal blocks have a size of 24 diagonal blocks diagonal size 24, off diagonal size 24 because the matrix size is 48 diagonal size is 24 it is incidental, then it is exactly like this suppose it is 2 4 and 4 2, then the total will be 20 and so on.

Now, the number of ones in this is given by 6 plus 4 10 plus 3, 13, 16, 21, 23, 25, the number of ones in the matrix is 25 next check that again, 4 plus 3 7 plus 2, 9, 12, 15, 19, 22 25, 25 ones are present. Now in an ideal situation all the ones should be present in the diagonal blocks and all the 0 should be present outside the diagonal blocks. So, in an ideal situation there should be 24 ones inside the diagonal blocks and 24 zeros outside the diagonal blocks. So, now out of the 25 ones that we have 4 of them are lying outside. So, 21 are lying in the diagonal blocks.

(Refer Slide Time: 26:48)



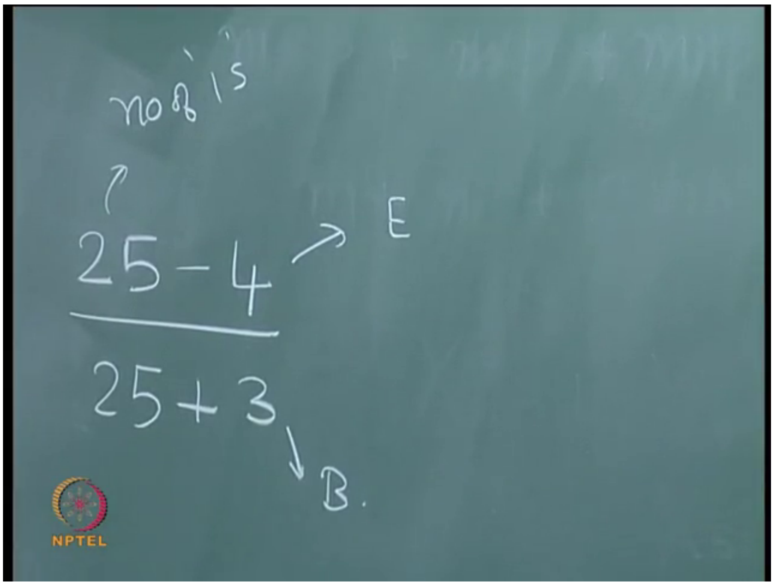
So, 21 are in the diagonal blocks out of a maximum 24 that the diagonal blocks can accommodate correct. This is for the diagonal the off diagonal size is 24. So, off diagonal size is 24 ideally all the 0 should come to the off diagonals, now how many zeros are there in the off diagonals? 20 are there because 4 ones are there, so plus 20 by 20 this is off diagonal.

Now, grouping efficiency is actually these efficiencies, the efficiency within the block efficiency outside the block. So, efficiency within the block is 21 by 24 into 100 percent this is 20 by 24 into 100 percent. So, if we add these 2 we get something out of 200. So, divide by 2 will give an efficiency as a percentage on 100. This is a very simple definition of the efficiency of the solution. So, it may repeat it again, now starting with a given matrix we have got a solution which has machines 1 3 4 in one group 2 5 6 in another group similarly parts 1 4 5 6 in one group 2 3 7 8 in another group. So, we have created 2 blocks diagonal blocks which are shown in this picture. So, the size of the diagonal blocks or 3 into 4 plus 3 into 4. So, size of the diagonal block is 24. Ideally we want all the ones to be inside the diagonal blocks and we observe that out of the 25 ones which are there in the original matrix, 21 of them are there in the diagonal block therefore, the diagonal efficiency is 21 by 24. Ideally we want all the zeros to be outside the diagonals. So, the outside diagonal space is another 24, it need not be equal to this total size is 48 this is 24. So, this is 24.

Now, we want all the zeros 24 zeros outside, but we observe that there are only 20 zeros outside because 4 ones are occupying the outside space. So, this efficiency is 20 by 24. So, total efficiency is this plus this by 2, which will give us an efficiency for 100 percent. Sometimes we may want to say that this has a higher weightage this has a lower weightage, in such a case we could say that this will be some q time this plus 1 minus q times this, where q is between 0 and 1 in such case we do not have to divide by 2 if q equal to 0.5 we could say point 5 equal weightage for both. So, this is called grouping efficiency, this is another way to measure the goodness of a block diagonal structure or goodness of a solution.

Another term that we have used is called grouping efficacy, now grouping efficacy is like this the total number of ones.

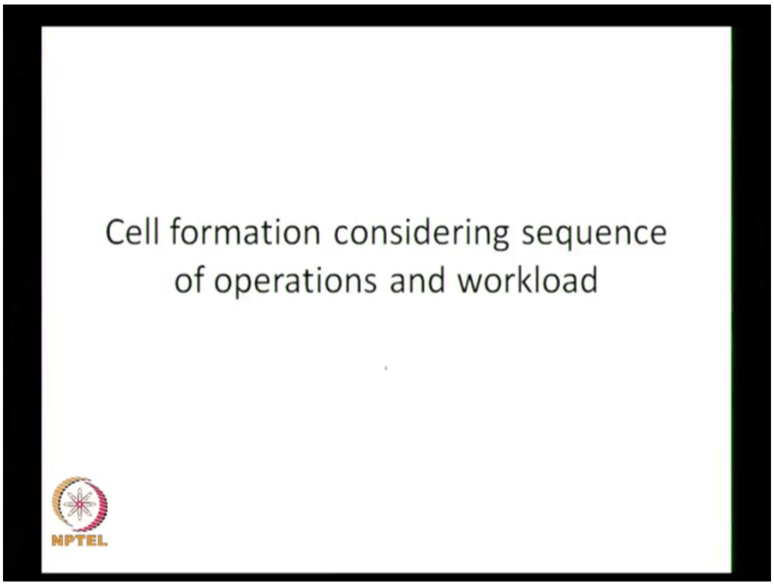
(Refer Slide Time: 30:52)



In this matrix is 25, the ideal solution is when all the ones are packed inside and all the zeros go outside, there is no 0 inside there is no 1 outside that is a ideal situation that would give us 100 percent efficiency.

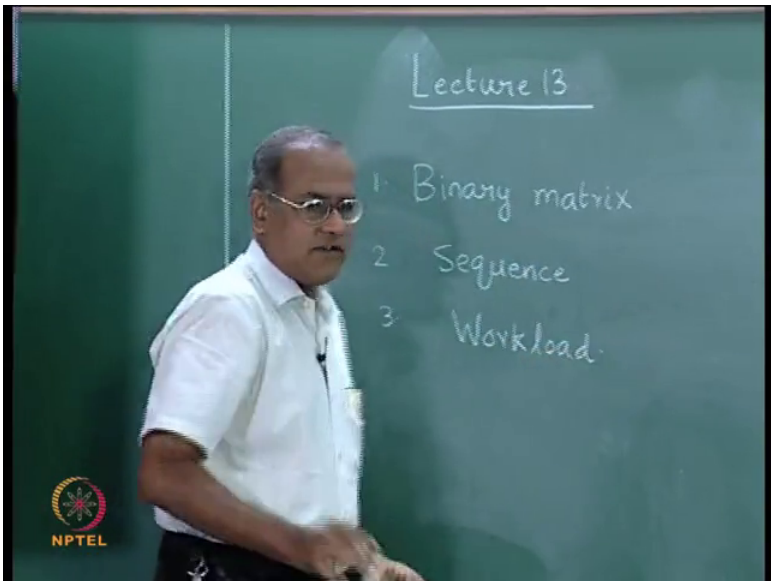
But now some ones are outside some zeros are inside. So, how many are inside 25 minus 4 are inside. The other 1 is by 25 plus 3; this is the number of ones this is E this is B. So, this definition is from Kumar and Chandrasakarn who gave this definition. So, ones minus exception by ones plus blank would give us. In the ideal case both these should be 0 therefore, we will have 100 percent efficiency. So, these are some of the common measures that are being used to asses or compare different solutions in the context of this.

(Refer Slide Time: 32:44)



Now so far we have seen several algorithms, which provide us solution to cell formation problem.

(Refer Slide Time: 33:06)



When we have a binary matrix, but we have already seen that the binary matrix does not consider operating sequence and does not consider work load.

First let us understand why these 2 are important and then see how we can look at algorithms that handle this requirement.

(Refer Slide Time: 33:49)

	1	2	3	4	5	6	7	8
1	1			2	1	1		
2		1				4	1	1
3	3			3		2		
4	4				3	3		
5		2	**2**		2		2	3
6	**2**	**3**	1	1			3	2

1, 2, 3 --- 1, 4, 6

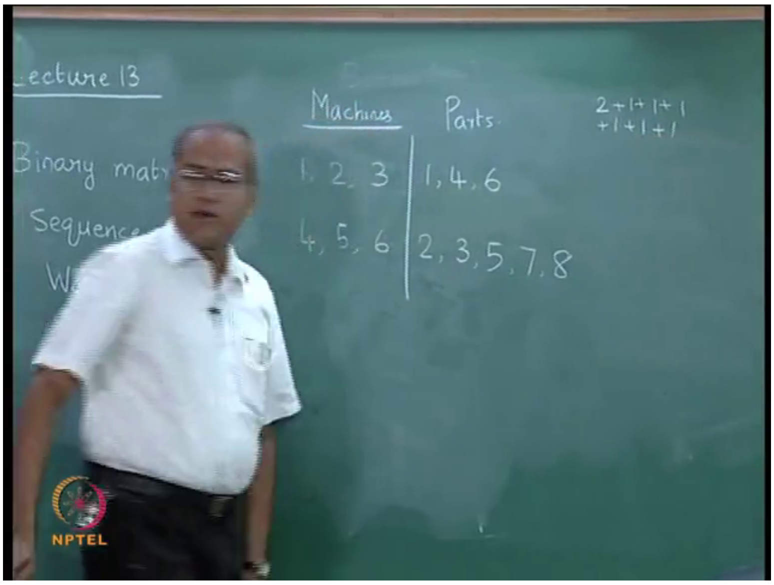
4, 5, 6 --- 2, 3, 5, 7, 8

10 moves

So, let us go back to an incidence matrix here, now let us look at this incidence matrix with 6 rows, 6 machines and 8 parts. Now the difference is this incidence matrix captures what is called the sequence. Now part number 1 visits machine 1 3 4 and 6, but visits in a particular sequence. So, it visits machine 1 first, then it visits machine 6, then it visits machine 3 and then it visits machine 4. Similarly part 2 visits 2 5 and 6 in a certain sequence.

Now, let us understand the idea of intercell moves, particularly in the context of the sequence. So, in order to do that let us consider 2 machine groups and then try to understand how many intercell moves are created particularly when sequence is also taken into account. So, let us look at a solution with 1 2 3 has 1 group machines.

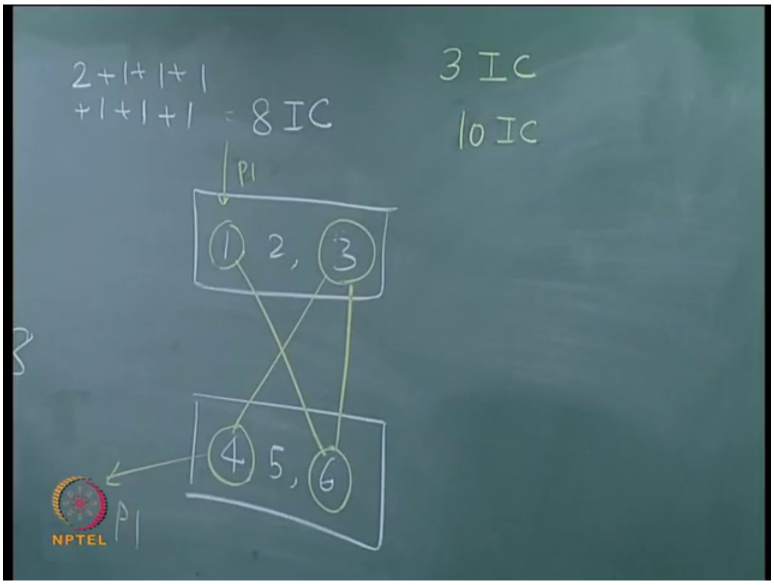
(Refer Slide Time: 35:16)



This may not be the optimal configuration, but this is just 1 configuration that we are considering to explain some simple things. Now let us allocate the parts using our part assignment rule. So, part 1 requires machines 1 3 4 and 6. So, part 1 requires machines 1 3 4 and 6. So, it can go to this it can go to this 1 3 4 and 6. So, part 1 goes here part 2 requires 2 5 and 6. So, part 2 will go here. Mean while it is also calculate the intercell moves as we calculate. So, part 1 requires 1 3 4 and 6, 1 3 4 and 6. So, 2 intercell moves part 2 requires 2 5 and 6, 5 6 2 1 intercell move. Part 3 requires 5 and 6 no intercell move part 4 requires 1 3 and 6, part 4 requires 1 3 and 6 plus 1 intercell move part 5 requires 1 4 and 5. So, plus 1 intercell move; part 6 requires 1 2 3 4. So, part 6 requires 1 2 3 4 part 7 requires 2 5 and 6 part 7 requires 2 5 and 6, part 8 requires 2 5 and 6, part 8 2 5 and 6.

So, this is how the parts would be assigned. So, this matrix is different from the 6 by 8 that we have seen or it assign parts this way for a given configuration which is like this it is different in the sense that part sequences are also added.

(Refer Slide Time: 37:53)



So, if we have these 2 machine groups, we will have this part allocation with 8 intercell moves. But let us try and map the intercell moves let say that here is a group 1 2 and 3, here is a machine group 4 5 and 6. Now let us trace all the parts; part 1 the visit is 1 6 3 and 4. So, part 1 comes here 1 6 3 4 and it goes. So, this is P 1.

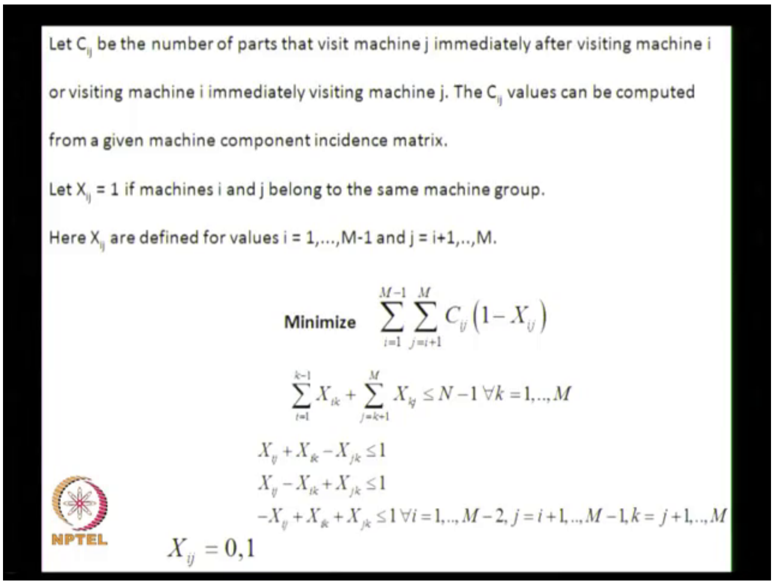
It first comes to 1, in this calculation we looked at 1 6 3 and 4; we said 1 and 3 are here 4 and 6 are here there are 2 intercell moves or if you go back, it starts with 1 there is 1 intercell move to go from 1 to 6, there is a second intercell move to come from 6 to 3 and there is a third intercell move to come from 3 to 4 and then it goes whereas, while we have written 2 here, there are actually 3 intercell moves. The moment we consider the sequence the exception there are 2 exceptions from the point of view of the off diagonals, but in terms of number of intercell moves there are 3 intercell moves.

So, in a similar manner if we look at all of them instead of 8 intercell moves, we will get 10 intercell moves. So, the cell formation considering intercell moves should take into account this aspect and not treat a visit of a machine outside the group as an intercell move.

There are actually 3 intercell moves instead of 2; interestingly the moment we bring sequence into consideration. This is only notational we actually do not need to do part allocation this is notational, because the root now becomes important and not which machine group it is assigned. So, if you really want to minimize intercell moves considering sequence data, then we should solve another optimization problem and not the optimization problem of boctor.

The boctors formulation does not consider sequence data. So, boctors formulation would treat this as 2 intercell moves and not 3. If we want to develop similarity coefficient based algorithms, then similarity coefficients have to be now redefined by taking to account the order of visit of the machines. So, we look at both these aspects now has to how we formulate another problem to do this.

(Refer Slide Time: 41:22)



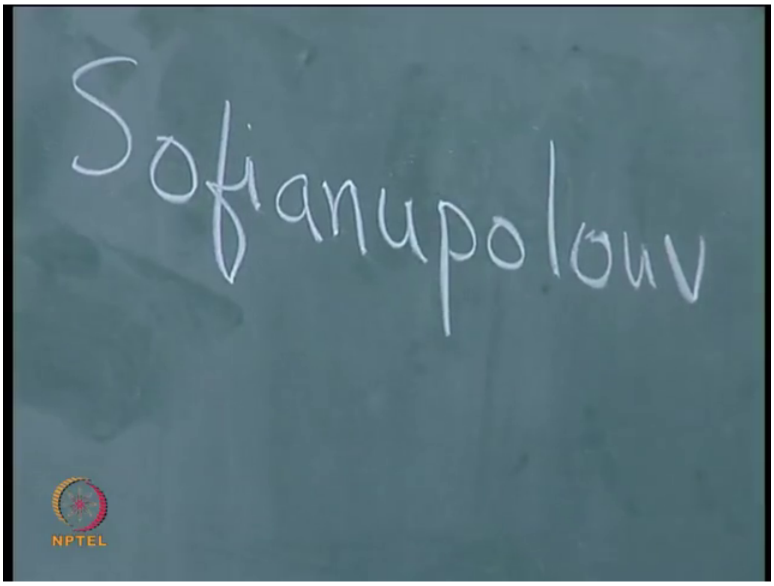
So, we first look at one optimization algorithm and then we look at one similarity based algorithm to solve this.

Now, the optimization algorithm goes like this. Let Cij be the number of parts that visit machine j immediately after visiting machine I or visiting machine I immediately after visiting machine j. So, this is a definition of C i j. So, it is a number of parts that visit j immediately after visiting i or visit i immediately after visiting j. So, in our matrix if we look at machines 1 and 2 the number of parts that visit 2 immediately after 1 or 1 immediately after 2. So, part 1 does not do that, part 2 does not do that, part 3 does not do that 6 is a only part which can be a candidate because it visits both the rest of them do not visit them 6 also does not visit 1 immediately after 2 or 2 immediately after 1 therefore, c 1 2 will be 0.

If we consider 3 and 4 you realize part 1 does that. It visits 4 machine number 4 immediately after visiting machine number 3; part 6 visits machine number 4 immediately after visiting machine number 3 therefore, c 3 4 will be 2 like this the Cij can be calculated. If you look at 5 and 6 you see now that part 2 visits 6 immediately after visiting 5 part 3 visits 5 immediately after visiting 6 part 7 visits 6 immediately after visiting 5 part 8 visits machine 5 immediately after visiting machine 6. So, C 5 6 will be 4.

Like this we can compute the Cij values. So, Cij values can be computed from a given machine component incidence matrix now the formulation is as follows. Let Xij equal to 1 if machines i and j belong to the same group, this formulation is given by Sofianupolouv formulation.

(Refer Slide Time: 44:20)



So, this assumes that Xij equal to 1 if machines i and j belong to the same group. Now Xij are defined for values i equal to 1 to M minus 1 and j equal to i plus 1 to M. If there are 6 machines in our incidence matrix i is equal to 1 to 5 j is equal to 2 to 6 and i not equal to j. So, automatically we will assume that C i i is infinity is a large value. So, minimization problem. So, you do not want that C i i will be a large value it would not be 0 it will be a large value.

So, you will define something like x 1 2, 1 3, 1 4, 1 5, 1 6, 2 3, 2 4, 2 5, 2 6, 3 4, 3 5, 3 6, 4 5, 4 6 and 5 6 that will be the definitions of Xij minimize sigma Cij into 1 minus x I j.

So, what does the objective function do if Xij is 1 which means i and j belong to the same group 1 minus Xij is 0 or take a situation where Xij is 0 which means i and j belong to different groups they do not belong to the same group Xij is 0. So, they do not belong to the same group, they belong to different groups.

So, when they belong to different groups what does Cij represent? Cij represents the number of parts that will visit either i immediately after j or j immediately after i. So, when i and j belong to different groups there will have to be Cij intercell moves not exceptions because we are counting both visiting i immediately after j as well as j immediately after i. So, there will be so many intercell moves. So, if Xij is 0 then 1 minus Xij will be 1 and that many intercell moves will be created.

Now, we want Xij such that we want to minimize that many intercell moves therefore, we minimize Cij into 1 minus X i j. So, Cij into 1 minus Xij will tell us the number of intercell moves and that is to be minimized. So, that is the objective function the there are sets of constraints the first set first constraint restricts the number of machines in a group. Plus once again it can give you a solution where it puts all the machines in one group and say that I have 0 intercell moves; plus if you have a trivial solution with all the machines in one group and all the parts in one group then there is no intercell move. So, that has to be avoided therefore, there has to be a restriction on a cell size.

Now, this capital n is the restriction on the cell size 6 machines or 7 machines or whatever is the cell size restriction now what is this constraint? This summation is only for k, there is a now actually k equal to 1 to M also represents number of groups in this case notionally where at the end it will k m this is the maximum number of groups, but then it will give us solutions with fewer number of groups. So, this will restrict the cell size to M minus 1 how does that happen.

Suppose I take a particular case, suppose I take machines 1 and 2. Now if x 1 2 equal to 1, x 1 3 equal to 1 then I have 1 2 and 3 equal to 1 they are in the same group. So, what we do is, for a particular k X i k. So, if I say 1 2 equal to 1 then I look at all the ones with 2 what are the other things that 2 is connected; with in 1 3 is 1 what are the other things 3 is connected with. So, all those will finally, belong to this cell. So, that size should be less than equal to n minus 1 n minus 1 comes because of the i that i considered, The starting machine the addition should be less than or equal to n minus 1 plus this machine will give a size of n. So, this constraint will give us a size of n for each cell now these constraints the 3 constraints will ensure that 1 and 2 belongs to a cell, 2 and 3 belong to the same cell, then 1 and 3 should also belong to the same cell.

If 1 and 2 belong to the same cell 2 and 3 belong to different cells then 1 and 3 should belong to different cells. So, all these conditions have to be mapped because by definition Xij is machines i and j belonging to the same cell. So, we need constraints to take care of that care full look at these constraints will tell us that these constraints map that. If i and j belong to the same cell, j and k belong to the same cell then i and k should belong to the same cell. So, if Xij is 1, x j k is 1, then x I k has to be 1. If Xij is 1, X j k is 0 then X i k has to be 0. So, all those are mapped by these constraints and Xij is binary. Now this is the formulation this formulation can be solved as it is there is no absolute value a term there is nothing like that. So, this can be solved as it is and this can be applied to our matrix. So, we will see the optimum solution to this problem as well as a heuristic solution to this in the next lecture.