Modified CI-Algorithm - A Heuristic for the GT-Problem for the Case of Multiple Process Plans

Christa BOYER

Forschungsbericht/
Research Memorandum No. 291

Jänner 1992

ABSTRACT

The Group Technology Problem deals with the decomposition of a manufacturing system into smaller subsystems which are easier to manage and organize. It is also known as the group technology concept and applied usually to identify machine cells and corresponding part families that are isolated in such a way that no flow of parts between those cells has to take place. This problem can be solved by the use of clustering algorithms. Kusiak et al. developed some heuristics that are less time consuming than the common optimization techniques. A modification of one of those heuristics shall be presented in this paper. The modification is necessary in order to apply the algorithm to some special form of group-technology problems: for the case of multiple process plans.

ZUSAMMENFASSUNG

Das Group Technologie Problem beschäftigt sich mit der Dekomposition von Fertigungssystemen in kleinere Subsysteme, die einfacher organisiert werden können. Es wird auch als Gruppentechnologie Konzept angesprochen und wird zumeist zur Identifikation von Maschinenzellen und korrespondierenden Teilefamilien eingesetzt. Diese sind so gestaltet, daß kein Transport der Teile von einer in eine andere Zelle notwendig ist. Das Problem ist mit Hilfe von Clustering-Algorithmen lösbar. Kusiak et al. entwickelten Heuristiken, mit denen solche Probleme in kürzerer Zeit gelöst werden können als mit Optimierungstechniken. In der vorliegenden Arbeit wird eine Modifikation einer dieser Heuristiken präsentiert. Diese Modifikation ist notwendig um den Algorithmus auf eine besondere Form von Group-Technologie-Problemen anwenden zu können: auf den Fall von mehrfachen Prozeßplänen.

1.INTRODUCTION

The Group Technology Problem deals with the decomposition of a manufacturing system into smaller subsystems which are easier to manage and organize. It is also known as the group technology concept and applied usually to identify machine cells and corresponding part families that are isolated in such a way that no flow of parts between those cells has to take place. This problem can be solved by the use of clustering algorithms. Kusiak et al.[1)] developed some heuristics that are less time consuming than the common optimization techniques. A modification of one of those heuristics[2)] shall be presented in this paper. The modification is necessary in order to apply the algorithm to some special form of group-technology problems: for the case of multiple process plans.

2.DEFINITION OF THE PROBLEM

As indicated before the GT-Problem in case of multiple process plans is a special form of the group technology problem (for the formulation of the GT-Problem see Kusiak p.206 - 246). In this problemspecification for one or more parts, that are to be manufactured on several machines, exist at least two different production plans, only one of which is to be selected. These production plans differ in a way that they do not involve exactly the same machines. The problem shall be illustrated by the following example which will later on also be used to demonstrate the algorithm.

Example:

part number	1			2			3		4		5		6	
process plan	1	2	3	4	5	6	7	8	9	10	11	12	13	14
machine 1	1				1		1	1					1	
2		1	1						1	1				
3	1					1						1		1
4				1		1		1					1	
5	1	1								1	1	1		1
6			1	1				1		1	1			1
7			1		1	1	1		1					1
8				1	1		1					1		

In this example six different parts are to be manufactured on eight machines, where part one and two can be manufactured in three different ways and parts three to six each have two alternative process plans. The problem is to decide, if there is a way to produce these parts in separated machine cells. For example consider part three and six: in a machine cell (1,4,6), that means the cell consists of the machines with number one, four and six, those parts can be produced by using the process plans eight and thirteen. Therefore these parts and machines form a so called part family (3,6) and corresponding machine cell (1,4,6). But in this case it can easily be seen that in order to produce part 2 either machine 8 or machine 7 and 8 or machine 3 and 7 have to be added to the machine cell. By looking at the other parts it becomes apparent that none of these possibilities leads to a reasonable decomposition of the production system.

As additional decision variable a cost-vector holding the production costs associated with each process plan can be considered. Kusiak states that these costs can be minimized by solving the mathematical programming formulation of the problem presented in the next part of the paper. The heuristic described in the fourth section assumes that all process plans cause the same production costs. The only way to take account of different costs is to exclude the most expensive process plans and compare the result of the heuristic with that of the complete problem. This might lead to the conclusion that it is better to produce one part in several machine cells, and accept the corresponding flow-cost (cost for the transportation of the part from one machine cell to another), than to use a very expensive process plan, that fits in one cell.

3. MATHEMATICAL PROGRAMMING FORMULATION - GENERALIZED P-MEDIAN MODEL

Kusiak[3] formulated the following programming model to solve the problem described above:

objective function:

$$\min \sum_{i=1}^{n} \sum_{j=1}^{n} d_{ij} x_{ij} + \sum_{j=1}^{n} c_j x_{jj}$$

constraints:

$$\sum_{i \in F_k} \sum_{j=1}^{n} x_{ij} = 1 \qquad \text{for all } k = 1,\dots,l \qquad (1)$$

$$\sum_{j=1}^{n} x_{jj} \le p \qquad (2)$$

$$x_{ij} \le x_{jj} \qquad \text{for all } i = 1,\dots,n,\ j = 1,\dots,n \qquad (3)$$

$$x_{ij} = 0,1 \qquad \text{for all } i = 1,\dots,n,\ j = 1,\dots,n \qquad (4)$$

notation:

x_{ij}.......indicates if process plan i is in the same part family as process plan j ($x_{ij}=1$ means that process plan i and j are selected and belong to the same part family)

n.........total number of process plans

F_k.......set of process plans for part number k, k=1,...,l,

p.........the maximum number of part (process) families

d_{ij}.......distance measure between process plans i and j (e.g.: Hamming Distances)

c_i........production cost of process plan i

Constraint (1) ensures that for each part exactly one process plan is selected. Constraint (2) imposes the upper bound on the number of part families. Constraint (3) imposes the condition that a process plan can only be manufactured in an existing part family. A short remark shall be made on the objective function: in this formulation the production costs are underestimated, as only the x_{jj} but no x_{ij} for i not equal j are considered. Therefore it is suggested that the objective function should be modified as follows:

$$\min \sum_{i=1}^{n}\sum_{j=1}^{n} d_{ij}x_{ij} + \sum_{i=1}^{n}\sum_{j=1}^{n} c_i x_{ij}$$

This formulation considers the production costs of all selected process plans.

There is one important disadvantage associated with this programming formulation: it is very time consuming, especially the task of calculating the distances between any two process plans. For ten process plans fortyfive distances, and for a more realistic number of onehundred alternative plans 4950 distances have to be calculated. To overcome this problem a heuristic approach has been developed and will be presented in the following section. There are two more disadvantages of the mathematical formulation:

- It is not ensured that one machine can only be used in one machine group.
- It is not possible to restrict the maximum number of machines per cell. Therefore the algorithm formulated by Kusiak will always produce a solution, in the worst case only one large machine cell including all selected part families will be identified.

The heuristic includes constraints that prevent such results. The maximum number of machines can be specified by the user, and each machine can only be linked to one cell.

4.EXTENSION OF THE CI-ALGORITHM

The modified CI-Algorithm can identify machine-cells and corresponding part families, provided that they exist. There need not necessarily exist a solution that includes only clear cut cells and families, but the heuristic can also produce the result that only some parts can be manufactured in the identified machine groups[4)].

4.1.THE ALGORITHM

In the following the algorithm will be presented step by step completed by an explanation of each operation.

INPUT:

The modified CI-Algorithm requires the following input data:
- the process plan-machine incidence matrix
- the number of parts with the additional information, which process plans correspond to which parts
- the maximum number of machines per cell

Step 0: set iteration=0

Step 1: set k=iteration+1

Step 2: Calculate a vector e_j which counts the number of entries in the incidence matrix for each machine j, and find the maximum of the elements of vector e_j. If there is more than one maximum calculate the similarities (s_{ij}) between these machines and select one of those that have maximum similarity. This machine is the first element of the machine cell k.

s_{ij} = number of equal entries for machine i and j divided by e_j

Step 3: Calculate the similarities (s_{ij}) between machine i selected in step 2 and all other machines.

Step 4: Choose that machine j with maximum similarity s_{ij}. If there is more than one maximum, select the machine with the higher counter (greater number of same entries as machine i). Add the now selected machine to machine cell k.

Step 5: Check how many parts can be manufactured in the preliminary machine cell. If there are at least two, accept the machine cell and corresponding part family and go to step 7. If it is less than two, go to step 6.

Step 6: Check if the number of machines in the preliminary machine cell is already equal to the maximum number of machines per cell. If this is true for the first time, return to step 1 and set e_i, where i is the machine selected first, to 0.

-If this is true for the second time: Stop

-If this is false, calculate the similatities sn_{ij} between the last selected machine and all remaining machines.

set $s_{ij} = s_{ij} + sn_{ij}$

Go to step 4

Step 7: Check if all the selected machines are needed to produce the parts in the part family k. If one or more are not necessary, drop those machines and put them back. Go to step 8.

Step 8: Reduce the incidence matrix in the following way:

a: Remove all machines that have been selected in the identified machine cell.

b: Remove all process plans of parts that are included in the part family k.

c: Remove all remaining process plans that would need one of the machines, that now have been removed and remember them as process plans $h_{p,k}$, where p stands for the process plan number and k indicates to which machine cell $h_{p,k}$ corresponds.

If the remaining matrix still includes producable process plans then go to step 1,

if not go to step 9.

Step 9: Check if all parts are represented by one of their process plans in one part family.
If this condition holds then STOP,
otherwise consider the machine cells and the corresponding process plans $h_{p,k}$ related to the not included parts. Check if it is possible, to accept one or more additional machines without offending the condition of maximum number of machines per cell and to enlarge the part family.
If it is possible, enlarge the machine and part group and repeat step 9,
if not then STOP.

STOP: The algorithm cannot produce any better solution.

4.2. DISCUSSION OF THE ALGORITHM

The idea of starting with a machine, that has the most entries in the matrix is based on the assumption that such a machine will be the center of a machine cell, and will probably be necessary for the production system. In case that this starting assumption does not produce a solution, that is, no machine cell can be identified, a second attempt excluding the first selected machine from the considered starting machines is made. In this form of the algorithm no more such returning steps are allowed. If necessary, this condition could be changed without further problems.

A few words shall be added to the similarity measure used. There exist a lot of similarity measures for binary data. The one applied here is a slight modification[5)]. The idea is, that for the selection of the machines only the number of equal entries is important and not that of equal non-entries, and that the similarities have to be weighted by the total number of entries for each machine. Otherwise a specialized machine, that might be of great importance for the cell, but has only a small number of entries, would only have very little probability of being selected.

Another point to be mentioned is that for the selection of the machine, which should be added to a preliminary cell, that already includes at least two machines, the sum of the similarities of all these machines to all the others is considered. The idea is to make sure to choose the machine, that has greatest similarity to the whole cell.

As already pointed out this algorithm is a modification of the CI-Algorithm of Kusiak. The heuristic will not always produce an optimal solution. The largest problems tested (the problems tested and the results are described in the appendix) included 25 machines and 41 process plans. The solution produced was optimal[6)] for the case where a solution including all parts existed. The algorithm also found a solution for the case where two parts could not be produced in the selected machine cells. Solving such problems by Kusiak's programming formulation will either lead to the identification of one single machine cell or to the selection of one machine for more than one cell. The heuristic also found a solution for the other problems tested. In some cases it is not possible to prove that the solution is optimal, because Kusiak's programming formulation produces results that violate a condition of the heuristic: one machine is selected for more than one cell.

A basic assumption of the heuristic is that the solution only includes part families which comprise at least two different parts. It might occure, but it is not very likely, that one part can be manufactured by more than one process plan in the same machine cell, but this would not influence the condition of a minimum of two parts per family.

It should be stressed that the algorithm cannot only find a solution when there exist just separable families and cells, but also in such cases where only a part of the matrix can be decomposed. It can also be used to test the improvement that can be reached by considering some additional machine(s), or if it is possible, to keep a good solution, if one machine is eliminated. These problems can be checked by applying the algorithm to a modified incidence matrix.

In case that the solution produced is not sufficient, the result might improve by changing the maximum number of machines per cell and applying the algorithm again.

In the following each step of the algorithm shall be illustrated by solving the example given before.

4.3.ILLUSTRATING EXAMPLE

Input:

part number	1			2			3		4		5		6	
process plan	1	2	3	4	5	6	7	8	9	10	11	12	13	14
machine 1	1				1		1	1					1	
2		1	1						1	1				
3	1					1						1		1
4				1		1		1					1	
5	1	1								1	1	1		1
6			1	1				1		1	1			1
7			1		1	1	1		1					1
8				1	1		1					1		

maximum number of machines per cell: 4

Step 0: iteration=0

Step 1: k=1

Step 2: calculate e_{ij}: [5 4 4 4 6 6 6 4]

max(eij)=6; corresponding to machines 5, 6 and 7

similarities of these machines:

	6	7
5	3/6	1/6
6	---	2/6

select machine number 5 or 6 arbitrarily -> 5

Step 3: calculate similarities of machine 5 to all others

j	1	2	3	4	6	7	8
s_{ij}	1/5	2/4	3/4	0/4	3/6	1/6	1/4

Step 4: select machine 3

Step 5: no parts can be produced in the preliminary machine cell (3,5)

Step 6: number of machines in cell (3,5) is 2 -> less than 4

j	1	2	4	6	7	8
sn_{ij}	1/5	0/4	1/4	1/6	2/6	1/4
s_{ij}	2/5	2/4	1/4	4/6	3/6	2/4

Step 4: choose machine 6

Step 5: part number 5 can be produced in the preliminary machine cell (3,5,6); less than 2 parts

Step 6: number of machines in cell (3,5,6) is 3 -> less than 4

j	1	2	4	7	8
sn_{ij}	1/5	2/4	2/4	2/6	1/4
s_{ij}	3/5	4/4	3/4	5/6	3/4

Step 4: choose machine 2

Step 5: parts number 1, 4 and 5 can be produced in the machine cell (2,3,5,6) by process plan 2, 10 and 11 -> more than 1 part

Step 7: machine number 3 is needed for none of the selected process plans and is therefore eliminated from the cell; the cell now consists of machines 2, 5 and 6, and the corresponding part family includes part number 1, 4 and 5.

Step 8: reduction of the incidence matrix

a: remove column 2, 5 and 6 from the matrix

b: remove the process plans 1, 2, 3, 9, 10, 11, 12

c: remove the process plan number 4, because it would need machine 6, which has already been selected, and also process plans 8 and 14. Remember them as $h_{4,1}$, $h_{8,1}$ and $h_{4,1}$ respectively. Process plan 6 is still a feasible plan, because machine number 3 has been put back.

The remaining matrix still includes producable process plans.

new matrix:

plannumber		2		3	6
process plan		5	6	7	13
machine	1	1		1	1
	3		1		
	4		1		1
	7	1	1	1	
	8	1		1	

Step 1: k=2

Step 2: e_{ij}: [3 1 2 3 2]

$\max(e_{ij})=3$ -> select arbitrarily machine 1 or 7; e.g. 1

Step 3: calculate the similarities of machine 1 to all others

j	3	4	7	8
s_{ij}	0/1	1/2	2/3	2/2

Step 4: select machine 8

Step 5: no parts can be produced in the preliminary machine cell (1,8)

Step 6: number of machines in cell (1,8) is less than 4

j	3	4	7
sn_{ij}	0/1	0/2	2/3
s_{ij}	0/1	1/2	4/3

Step 4: choose machine 7

Step 5: two parts can be produced in the machine cell (1,7,8); the part family consists of parts 2 and 3

Step 7: all machines of the cell are needed

Step 8: reduction of the incidence matrix

a: remove column 1, 7 and 8

b: remove row 5, 6 and 7

c: remove process plan number 13 and remember it as $h_{13,2}$

The remaining matrix has no more rows

Step 9: part number 6 is not producable in the existing machine cells

Process plan number 14 has been remembered as $h_{14,1}$ (machine cell 1 consists of the machines number 2, 5, 6); it would need the additional machines 3 and 7, but machine 7 is no longer available.

Process plan number 13 corresponds to $h_{13,2}$ (machine cell 2 includes the machines number 1, 7 and 8); it needs the additional machine 4 which is still available. Therefore enlarge cell 2 and form the new cell (1,4,7,8) with the corresponding part family (3,4,6).

Step 9: all parts can be produced in the existing machine cells

SOLUTION MATRIX

part number	1	4	5	2	3	6
process plan	2	10	11	5	7	13
machine 2	1	1				
5	1	1	1			
6		1	1			
1				1	1	1
4						1
7				1	1	
8				1	1	

5. DISCUSSION

The algorithm presented can be used to solve a special form of Clustering problems. Its advantage is that it is not as time consuming as the solving of the mathematical programming model, and that it includes some reasonable constraints, which are not considered in Kusiak's problem formulation. On the other hand the algorithm cannot take into account that different process plans might cause different costs. The development of a modification of this algorithm, that could also consider costs, might be a interesting task and make the algorithm more useful.

FOOTNOTES

1) A. Kusiak: Intelligent Manufacturing Systems, Prentice-Hall, New Jersey 1990, p.206-246

2) Kusiak, p.222

3) Kusiak, p. 229

4) In that case the program - the implementation of the algorithm - would produce the result: 'the following parts cannot be manufactured in the machine cells given', followed by the identified machine cells and corresponding process plans.

5) Backhaus, Erichson, Plinke, Weiber: Mulitivariate Analysemethoden, 6.Auflage Springer, Berlin 1990, S118ff

6) The optimal solution of the problems were found by solving the mathematical programming formulation of Kusiak with XPRESS.

APPENDIX

Testproblem 1: 8 machines, 6 parts and 14 process plans; maximal number of machines per cell: 4

process plan											
part number	1			2		3		4		5	
process plan-numbers	1	2	3	4	5	6	7	8	9	10	11
	0	0	1	0	1	1	0	1	1	0	1
	0	1	1	1	0	0	1	0	0	0	0
	1	0	0	1	1	0	0	0	1	1	0
	1	1	0	0	0	1	1	1	0	1	0

Results:	machine cell	machines	parts	process plans
	1	1,3	2,4,5	5,9,11
	2	2,4	2,3	2,7

Testproblem 2: 8 machines, 6 parts and 14 process plans; maximal number of machines per cell: 4

process plan														
part number	1			2			3		4		5		6	
process plan-numbers	1	2	3	4	5	6	7	8	9	10	11	12	13	14
	1	0	0	0	1	0	1	1	0	0	0	0	1	0
	0	1	1	0	0	0	0	0	1	1	0	0	0	0
	1	0	0	0	0	1	0	0	0	0	0	1	0	1
	0	0	0	1	0	1	0	1	0	0	0	0	1	0
	1	1	0	0	0	0	0	0	0	1	1	1	0	1
	0	0	1	1	0	0	0	1	0	1	1	0	0	1
	0	0	1	0	1	1	1	0	1	0	0	0	0	1
	0	0	0	1	1	0	1	0	0	0	0	1	0	0

Results:	machine cell	machines	parts	process plans
	1	2,5,6	1,4,5	2,10,11
	2	1,4,7,8	2,3,6	5,7,13

Testproblem 3: 13 machines, 8 parts and 24 process plans;
maximal number of machines per cell: 5

	processplan																							
part number	1				2				3			4			5			6			7		8	
process plan-numbers	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	1
	0	1	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	0	0	0	0	1	0	0
	0	1	0	0	0	0	1	0	0	1	1	0	0	0	0	0	0	1	0	1	0	1	0	0
	0	0	0	0	0	0	1	1	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	1
	1	0	1	1	0	0	1	0	1	0	0	0	0	0	0	1	0	1	1	0	1	0	0	1
	1	1	0	1	0	0	0	0	0	0	0	0	0	0	0	1	1	0	1	0	0	0	0	0
	0	0	0	0	0	1	0	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0
	0	0	1	0	1	0	0	0	1	1	0	1	1	0	0	0	0	0	0	0	0	0	0	0
	0	0	0	0	1	0	0	0	0	1	0	0	1	0	0	0	0	0	1	1	0	0	0	0
	0	1	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	1	0	1
	1	0	0	0	0	0	0	1	1	0	1	0	0	1	1	0	0	0	0	0	0	0	0	1
	0	0	1	0	1	1	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	1	0	0
	0	0	0	0	0	1	1	0	1	0	0	0	0	1	0	0	1	0	0	0	0	0	1	0

Results:

machine cell	machines	parts	process plans
1	2,5,6,10	1,5,7	4,16,21
2	3,8,9	3,4,6	10,13,20
3	1,7,12,13	2,8	6,23

Testproblem 4: 10 machines, 8 parts and 23 process plans;
maximal number of machines per cell: 5

	processplan																						
part number	1				2				3			4			5			6		7		8	
process plan-numbers	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
	0	1	0	0	1	1	0	1	0	0	0	1	0	1	0	0	1	0	0	0	0	0	1
	1	0	0	1	0	0	1	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0
	0	0	0	1	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	1	1	1	0
	0	1	0	1	1	1	0	0	1	0	0	0	0	0	1	1	0	0	1	0	0	1	1
	0	1	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	1	1	0	1	0
	0	1	0	0	0	1	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1
	1	0	0	0	1	0	1	0	0	0	1	0	1	0	1	0	0	1	1	0	1	1	0
	1	0	1	0	0	0	1	1	0	0	1	1	1	0	0	1	0	1	0	0	1	0	0
	0	0	1	0	0	0	0	1	0	1	1	1	0	1	0	1	0	0	0	0	1	0	0
	0	0	1	0	0	0	0	0	0	0	0	1	1	0	0	1	0	1	1	0	0	0	0

Results:

machine cell	machines	parts	process plans
1	2,7,8,10	1,4,6	1,13,18
2	1,4,6	2,5,8	6,17,23
3	3,5,9	3,7	10,20

Testproblem 5: 25 machines, 14 parts and 41 process plans; maximal number of machines per cell: 7

	process plan																																								
part number	1				2				3			4			5			6			7			8			9			10			11			12		13		14	
process plan-number	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41
	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
	0	1	0	0	0	1	1	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	1	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
	1	0	1	1	0	1	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	1	0	0	0	0	0	0	1	0	1	0
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	1	0	1	0	0	0	0	1	0	0	0	1	0	0	1	1	0	0	1	0	0	0
	1	1	0	0	1	0	0	0	1	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0	1	0	1	0	0	0	0	1	0	0	0	0
	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1
	0	0	0	0	1	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	1	0	0
	0	0	1	0	0	0	0	0	0	0	1	0	0	0	1	1	1	0	0	0	0	0	1	0	0	1	0	0	0	0	0	1	1	0	1	0	0	0	0	0	0
	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	1	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0
	0	0	1	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	1	0	0	0	0	0	0	0	0	1	0
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0
	0	0	0	0	0	1	0	0	0	0	0	1	0	1	0	1	0	0	0	0	0	1	0	1	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0
	0	1	0	0	0	0	1	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	1	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	1	0	0	0
	0	1	1	0	1	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	1	0	1	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1
	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	1	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	1	0	0	0	0
	1	1	0	0	1	1	0	0	0	1	1	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	1	1	1	0	1	0	0	0	0	0	0	1	0
	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
	1	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	0	0	1	0	0	0
	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	1	0	1	1	0	0	1	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0
	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	0	0	0	1	0	0	0	1	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	1
	0	0	0	1	1	0	0	0	1	0	0	0	0	1	0	1	0	0	0	0	0	0	0	1	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	1	1
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	1	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0	0
	0	0	0	0	0	0	1	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	1	0	0

Results:	machine cell	machines	parts	process plans
	1	3,5,17,19	1,3,10	1,10,30
	2	6,12,15,22,23	4,8,14	14,24,41
	3	11,16,20,21,24	6,7,12	19,21,36
	4	2,7,9,13,25	2,9,13	7,28,39
	5	1,4,8	5,11	17,35

Testproblem 6: 25 machines, 14 parts and 41 process plans; maximal number of machines per cell: 7

	process plan																																							
part number	1				2			3				4			5			6			7			8			9			10			11			12		13		14
process plan-number	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41
	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
	0	1	0	0	0	1	1	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	1	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
	1	0	1	1	0	1	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	1	0	0	0	0	0	0	1	0	1	0
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	1	0	1	0	0	0	0	1	0	0	0	1	0	0	1	1	0	0	1	0	0	0
	1	1	0	0	1	0	0	0	1	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0	1	0	1	0	0	0	0	1	0	0	0	0
	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1
	0	0	0	0	1	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	1	0	0
	0	0	1	0	0	0	0	0	0	0	1	0	0	0	1	1	1	0	0	0	0	0	1	0	0	1	0	0	0	0	0	1	1	0	1	0	0	0	0	0	0
	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	1	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0
	0	0	1	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	1	0	0	0	0	0	0	0	0	1	0
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0
	0	0	0	0	0	1	0	0	0	0	0	1	0	1	0	1	0	0	0	0	0	1	0	1	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0
	0	1	0	0	0	0	1	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	1	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	1	0	0	0
	0	1	1	0	1	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	1	0	1	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1
	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	1	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	1	0	0	0	0
	1	1	0	0	1	1	0	0	0	1	1	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	1	1	1	0	1	0	0	0	0	0	0	1	0
	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
	1	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	0	0	1	0	0	0
	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	1	0	1	1	0	0	1	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0
	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	0	0	0	1	0	0	0	1	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	1
	0	0	0	1	1	0	0	0	1	0	0	0	0	1	0	1	0	0	0	0	0	0	0	1	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	1	1
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	1	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0	0
	0	0	0	0	0	0	1	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	1	0	0

Results: two parts (5 and 11) cannot be manufactured in the identified machine cells

machine cell	machines	parts	process plans
1	3,5,17,19	1,3,10	1,10,30
2	6,12,15,22,23	4,8,14	14,24,41
3	11,16,20,21,24	6,7,12	19,21,36
4	2,7,9,13,25	2,9,13	7,28,39