Hierarchical Classes Analysis for the Group Technology Problem

Gerrit K. Janssens* and Alain De Beuckelaer

University of Antwerp (RUCA), Business Information Systems, Operations and Logistics Management, Middelheimlaan 1, B-2020 Antwerp, Belgium

*To whom all correspondence should be addressed

ABSTRACT

A new approach, called the Hierarchical Classes Analysis (HCA) approach, is introduced as a policy instrument in a group technology (GT) context, in which machine cells have to be formed. As opposed to cluster analysis (CA) approaches, the HCA approach is concerned with finding a hierarchical representation of the machine (class) and part (class) structure simultaneously. The application of the HCA approach is applicable in the event that mutually exclusive machine cells exist or if bottleneck machines (and/or exceptional parts) prohibit the formation of mutually exclusive machine cells. The alternatives offer various ways of either duplicating one or more machines or allowing for intercellular part transfer. The decision maker chooses the best solution according to a specified objective. The procedure is illustrated with an optimization problem: minimize intercellular part transfer, given a budget constraint for duplicating machines.

KEYWORDS: Group technology, machine cell formation, cellular manufacturing system, hierarchical classes analysis.

I. INTRODUCTION

Group technology (GT) is aimed at decomposing a manufacturing system into subsystems that are easier to manage than the entire system. Its introduction has led to a number of advantages, such as reduction in production lead time, work in progress, labor, tooling, rework, scrap material, set up time, and order time delivery.[1]

In the design process, the Cell Formation Problem (also called the Group Technology problem) is mentioned as one of the major problems.[2] The GT problem considers the grouping of parts into part families and machines into

machine cells, and the assignment of part families to machines cells. In an ideal cell configuration, products are completely manufactured within one cell and are then transported to an assembly cell to be assembled with other parts. In this ideal case, no intercell transfers of parts are required. Intercell movements can be reduced at any time by duplicating machines, at an increased cost. Research into quantitative methods helping the solution of the GT problem deals mostly with one of two types of objective functions: either (1) minimize a measure of intercell transfer volume, or (2) maximize a measure of machine utilization.

The greatest potential for the application of GT lies in batch-type manufacturing, i.e., many different products produced in small lot sizes. The volume for any particular part is not high enough to require a dedicated production line, but the volume of a family of parts is high enough to efficiently utilize a cell.

Kusiak[1,3] considers two basic methods for solving the GT problem: classification and cluster analysis (CA). The former is used to group parts into part families based on their design features, e.g., by visual recognition of similarity of the geometric shape, or by assigning a code to each part on basis of features such as geometric shape, dimensions, type of material, or required accuracy. The latter method is more extensively discussed in the next section.

II. CLUSTER ANALYSIS APPROACHES FOR THE GT PROBLEM

CA (especially 1-dimensional CA approaches) focuses on the formation of machine cells, based on the calculation of a similarity coefficient (or a distance measure) between different groups of machines. From a GT perspective, 1-dimensional CA is able to focus on either the formation of machine cells or the formation of part families. In practice, however, there is an *a priori* basis for preferring the formation of machine cells over the formation of part families. If the implicit assumption of (some) homogeneity within machine cells,[1] is satisfied, CA is used to form a set of mutually independent machine cells, each able to process all operations of part families assigned to it. Only if a certain degree of mutual exclusiveness among machine cells exists can the different part families easily be derived from the CA solution.

In real-world applications, the assumption of mutual exclusiveness of either machine cells or part families is often violated. Not all components of a part family can always be processed in a single machine cell. The parts having operations in more than one cell are called *exceptional parts,* and the machines that process these operations are called *bottleneck machines.* These machines require either intercellular movements of the parts to be processed or duplication of the

bottleneck machines in the appropriate cells. The total number of intercellular movements and/or duplicated bottleneck machines should be minimized for practical and economic reasons.

Within the CA approach, three formulations of the GT problem are proposed: the matrix formulation, the mathematical programming formulation, and the graph formulation.[1,3] We will use the matrix formulation approach. In the matrix formulation, a part–machine incidence matrix $A = [a_{ij}]$ is constructed. The 2-dimensional matrix consists of binary a_{ij}-entries, where an entry 1 (0) indicates that machine i is used (not used) to process part j.

Heuristic formulations of the CFP typically are based on a part–machine incidence matrix, including which part is to be processed by which machine. Formulations based on the part–incidence matrix rely on assumptions that are open to criticism, e.g., they assume that each part requires an identical capacity requirement on each machine upon which it requires processing. Major deficiencies in using these methods have been observed in obtaining efficient cells in practical situations. Some options for improvement of the methods are: production volumes and capacity constraints should be included; tooling constraints should be considered; cell size constraints, sequence, and durations of operations should be accounted for.[4-6]

Whatever method the practitioner will use, it has to satisfy two requirements: it has to detect mutually separable machine cells, if they exist, and it has to determine the bottleneck machines (or the exceptional parts). The way in which mutually separable machine cells are identified depends on the clustering algorithm. Miltenburg and Zhang[7] evaluate 9 algorithms developed to solve the cell formation problem and compare them for their ability to produce good solutions to large problems (25–50 machines, 35–50 parts, and 10–20% entries in the incidence matrix having value 1).

One of the most popular CA methods is (agglomerative) hierarchical CA. Because of the hierarchical way in which clusters are formed, solutions are not at all unique. A solution for k machine cells can also be obtained from the solution of $k + 1$ machine cells when 2 machine cells are combined. The choice of the 2 machine cells to be combined depends on the defined criterion (e.g., the 2 machine cells with the largest similarity coefficient in King and Nakornchai[8]).

The use of hierarchical 1-dimensional CA to solve the GT problem is subject to some criticism. The 1-dimensional approach forces the aggregating process to be limited to the aggregation of *either* machines or parts. Because exceptional parts obstruct the formation of mutually exclusive machine cells, the part families cannot be derived from the different machine cells. Consequently, possible interrelationships between machine cells and groups of parts (or part families) cannot be discovered by means of these 1-dimensional CA approaches.

A partial solution to this problem is offered by 2-dimensional (hierarchical) CA approaches, the aim of which is to build a common categorical representation of the (similarity) structure of both sets of objects (e.g., machines) and sets of features (e.g., parts).[9] This common representation of both machines and parts enables the decision maker to detect mutually exclusive submatrices within the incidence matrix.

Recently, some models and algorithms have been developed to construct the common categorical representation of both objects and features. Objects and features, for our purposes, are instantiated as machines and parts. Examples are Arabie and Hubert's[10] revision of the Bond Energy Algorithm originally proposed by McCormick et al.[11] and Eckes and Orlik's[12] centroid effect method. All hierarchical (agglomerative) CA methods provide at most only 1 machine cell configuration for a given number of machine cells. Furthermore, these methods do not allow selection of the "best" solution (according to a "goodness" measure) among 2 or more solutions with an unequal number of machine cells.

In evaluating alternative machine cell configurations with equal or unequal number of machine cells, it is necessary to have more appropriate information concerning the strength of the interrelations or associations between, for example, a bottleneck machine and the part family to which exceptional parts belong. This kind of information is valuable whenever one has to decide on duplication of bottleneck machines or intercell transfers.

In the remainder of the paper, a new approach is introduced to cope with the above-mentioned criticisms. In Section III, the basic concepts of the hierarchical classes analysis (HCA) approach are presented. In Section IV, the HCA approach is illustrated with an example of mutually exclusive machine cells. In Section V, the HCA approach is illustrated with an example of overlapping machines and bottleneck parts. Finally, the decision support aspect is illustrated using an optimizing strategy.

III. THE HIERARCHICAL CLASSES ANALYSIS (HCA) APPROACH TO THE GT PROBLEM

III.A. Concepts and Basic Ideas

The HCA approach covers two requirements: (1) a common categorical representation of both sets of machines (machine cells) and sets of parts, and (2) information about the strength of the associations between a machine (or a machine cell) and a part (or a part family).

In HCA, a distinction is made between 2 types of classes: part classes and machine classes. A *part class* consists of parts processed on the same set of machines. Similarly, a set of machines used to process a common set of parts is called a *machine class*. The hierarchical approach is aimed at identifying subclasses in both part classes and machine classes, affording a variety of possible cell formations to the decision maker. An algorithm, called HICLAS (acronym for HIerarchical CLASses), is used to recover such an underlying deterministic hierarchical structure starting from the incidence matrix.

The basic data are stored as binary values in a 2-dimensional incidence matrix. The algorithm alternates between the rows (machines) and the columns (parts) of the incidence matrix to find the best-fitting row classes (sets of machines) and column classes (sets of parts) and their hierarchical relations, using an iterative Boolean regression technique.[13] In practice, the underlying (set-theoretical) structure of the actual incidence matrix is not completely deterministic. By allowing discrepancies between the actual incidence matrix and the recovered or modeled incidence matrix, the algorithm accounts for some measurement error. This approximation is done by minimizing the total number of discrepancies between both data matrices. The minimization process is discussed more extensively in later paragraphs. The set-theoretical relations in the HICLAS model are the best possible approximation of the relations that exist in reality.

In Section II.B, we focus on the possible set-theoretical relations among machines and parts. The Boolean decomposition and the set-theoretical decomposition will be discussed in Section V.

III.B. The Set-Theoretical Relations

Three set-theoretical relations are of importance in our HCA approach: the equivalence relation, the association relation, and the order relation. These relations will be discussed by means of an illustrative example.

Assume the following hypothetical machine–part incidence matrix A. The matrix is restructured in such a way that the relations are easily recognized.

The equivalence relation
Machines (m_i, $i = 1, \ldots, 8$) are considered equivalent if and only if they perform operations on identical sets of parts (p_j, $j = 1, \ldots, 7$). Likewise, parts are considered equivalent if and only if they share identical sets of machines in their routings. Sets of equivalent machines (parts) are called machine (part) classes. In Matrix 1, we can see that there are 4 machine classes {m1}, {m2,m3,m4},

MATRIX 1. Machine–part incidence matrix.

	p1	p2	p3	p4	p5	p6	p7
m1	1	1	1	1	1	1	1
m2	1	1	1	0	0	0	0
m3	1	1	1	0	0	0	0
m4	1	1	1	0	0	0	0
m5	0	0	0	1	1	1	0
m6	0	0	0	1	1	1	0
m7	0	0	0	1	1	1	0
m8	0	0	0	0	0	0	1

$A =$ (at left of the matrix rows)

{m5,m6,m7}, and {m8} and 3 part classes {p1,p2,p3}, {p4,p5,p6}, and {p7}. Note that machine (part) classes can consist of a single machine (part), e.g., {m1}, {m8}, and {p7}.

The association relation

A machine class M_k is associated to a part class P_l if and only if the parts in the part class P_l are processed on the machines in M_k. Likewise, a part class P_l is also associated to a machine class M_k if and only if the machines in the machine class M_k are used to process the parts in P_l. For instance, the machine class {m2,m3,m4} is associated to the part class {p1,p2,p3}, and vice versa.

The order relation

A machine class M_i is lower ordered than another machine class M_j if and only if its associated part class(es) is (are) a proper subset of the part classes associated with M_j. The machine classes {m2,m3,m4} and {m5,m6,m7} are lower ordered than machine class {m1} because their associated part classes ({p1,p2,p3} and {p4,p5,p6}, respectively) form a subset of the part classes associated to {m1} ({p1,p2,p3}, {p4,p5,p6}, and {p7}).

By introducing the order relation, the machine cells and part classes can both be partially ordered hierarchically. In Matrix 1, machine class {m1} is higher ordered than machine classes {m2,m3,m4} and {m5,m6,m7}, because the single machine m1 in the higher-ordered machine class {m1} is used to process all parts in the part classes associated with the lower-ordered machine classes {m2,m3,m4} and {m5,m6,m7}, in particular, {p1,p2,p3,p4,p5} and {p6}.

None of the machine classes {m2,m3,m4}, {m5,m6,m7}, and {m8} are higher ordered than any other machine class. They are called *bottom machine classes.* Also, none of the part classes {p1,p2,p3}, {p4,p5,p6}, and {p7,p8} are higher ordered than another part class, and they are called *bottom part classes.* Machine

(part) classes that are higher ordered than one or more other machine (part) classes are called *dominant machine (part) classes*. Each machine (part) class is either a bottom or a dominant machine (part) class.

IV. THE HCA APPROACH IN MUTUALLY EXCLUSIVE MACHINE CELLS

IV.A. Mutually Exclusive Machine Cells

A machine–part technology system consists of mutually exclusive machine cells if

1. All machines $m_i \in M$ belong to a machine class M_j in such a way that

$$\cup_j M_j = M \wedge M_j \cap M_k = \phi \ (\ \forall_j \neq k) \ ;$$

2. All parts $p_i \in P$ belong to a part class P_j in such a way that

$$\cup_j P_j = P \wedge P_j \cap P_k = \phi \ (\ \forall_j \neq k) \ ;$$

3. Each part/machine combination belongs to one and only one part class/machine class combination

$$\forall_{i,j,k,l} \ (\ p_i \in M_j \wedge m_k \in P_i \ ) : \neg \exists F \neq j, g \neq 1 : p_i \in M_r \vee m_k \in P_g \ .$$

Any clustering algorithm should be able to find a set of mutually exclusive machine cells within the machine–part incidence matrix. This simplified case is presented in order to explain our HCA approach.

In order to illustrate the use of the earlier-defined relations in a hierarchical way, we use an example represented by Matrix 2. The matrix is restructured in such a way that the reader easily can identify the machine cells.

Using the definition of equivalence, machine classes in this matrix are: {m1,m3,m4}, {m2}, {m5,m6}, {m7}, {m8,m9}, and {m10}. Part classes are {p1,p3}, {p2}, {p4}, {p5,p6}, {p7}, and {p8}.

A number of associations can be found among the identified machine and part classes, as shown in Table 1. The left side shows the associations from a machine viewpoint; the right side from a part viewpoint. The total amount of information on both sides is the same as the association relation is a symmetric relation.

MATRIX 2. Restructured incidence matrix.

		p1	p2	p3	p4	p5	p6	p7	p8
	m1	1	1	1	0	0	0	0	0
	m2	1	0	1	0	0	0	0	0
	m3	1	1	1	0	0	0	0	0
	m4	1	1	1	0	0	0	0	0
A =	m5	0	0	0	1	1	1	0	0
	m6	0	0	0	1	1	1	0	0
	m7	0	0	0	0	1	1	0	0
	m8	0	0	0	0	0	0	1	1
	m9	0	0	0	0	0	0	1	1
	m10	0	0	0	0	0	0	0	1

Table 1 contains the required information to identify a hierarchy of classes. Based on the definition of order relation, the pairs which can be relatively ordered to each other are shown in Tables 2 and 3. Both hierarchies in Tables 2 and 3 can be represented simultaneously by making use of a symmetric association relation between machine classes and part classes. When a machine (part) class is associated to a given part (machine) class, it is also associated to all its superordinate part (machine) classes.

In Figure 1, the minimal set of association relations is indicated by slashed lines.

Most real incidence matrices require a complex model to cover all associations. A simpler model implies relaxation of the types of relations in the data. If one allows for a minimum of discrepancies between the modeled incidence

Table 1. The Existing Associations (Matrix 2)

PART CLASS	MACHINE CLASS(ES)	MACHINE CLASS	PART CLASS(ES)
is associated to		is associated to	
{p1,p3}	{m1,m3,m4}	{m1,m3,m4}	{p1,p3}
	{m2}		{p2}
{p2}	{m1,m3,m4}	{m2}	{p1,p3}
{p4}	{m5,m6}	{m5,m6}	{p4}
{p5,p6}	{m5,m6}		{p5,p6}
	{m7}	{m7}	{p5,p6}
{p7}	{m8,m9}	{m8,m9}	{p7}
{p8}	{m8,m9}		{p8}
	{m10}	{m10}	{p8}

Table 2. Hierarchy of Machine Classes (Figure 2)

is higher ordered than	{m1,m3,m4} ↓ {m2}	{m5,m6} ↓ {m7}	{m8,m9} ↓ {m10}

Table 3. Hierarchy of Part Classes (Figure 2)

is higher ordered than	{p1,p3} ↓ {p2}	{p5,p6} ↓ {p4}	{p8} ↓ {p7}

matrix and the actual incidence matrix, much simpler models can be considered. Such simpler models will be constructed based on the HCA analyses.

The practitioner will easily find a simpler model for the matrix A. By ignoring the inefficiencies {m2,p2}, {m7,p4}, and {m10,p7}, 3 independent cells are easily recognized: {m1,m2,m3,m4}, {m5,m6,m7}, and {m8,m9,m10}. The matrix M corresponding to the simpler model is shown in Matrix 3.

Three independent part classes can be recognized: {p1,p2,p3}, {p4,p5,p6}, and {p8,p9,p10}. Association relations exist between {m1,m2,m3,m4} and
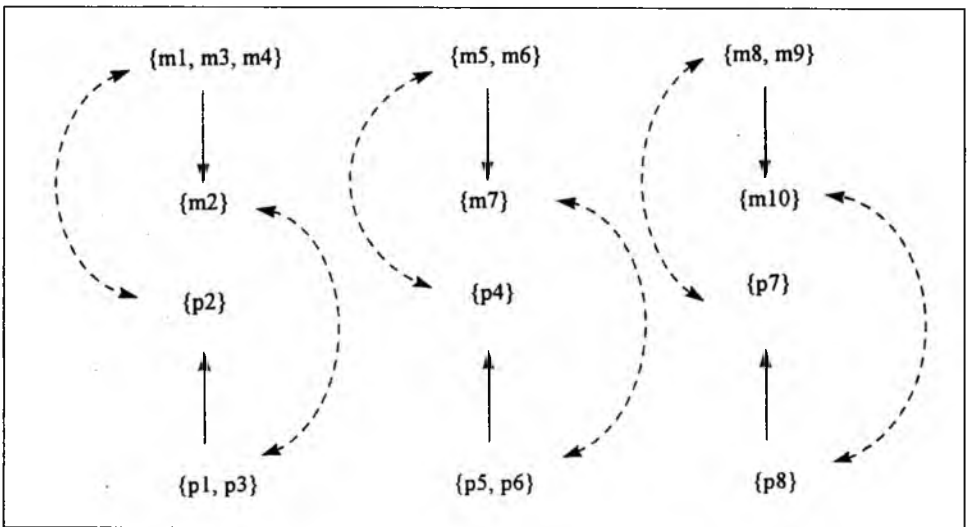

FIGURE 1.
Minimal set of association relations

MATRIX 3. Simplified incidence matrix.

		p1	p2	p3	p4	p5	p6	p7	p8
	m1	1	1	1	0	0	0	0	0
	m2	1	1	1	0	0	0	0	0
	m3	1	1	1	0	0	0	0	0
	m4	1	1	1	0	0	0	0	0
M =	m5	0	0	0	1	1	1	0	0
	m6	0	0	0	1	1	1	0	0
	m7	0	0	0	1	1	1	0	0
	m8	0	0	0	0	0	0	1	1
	m9	0	0	0	0	0	0	1	1
	m10	0	0	0	0	0	0	1	1

{p1,p2,p3}, between {m5,m6,m7} and {p4,p5,p6}, and between {m8,m9,m10} and {p7,p8}. No order relationship exists among the machine classes or part classes. In earlier defined terms, this means that all classes are bottom classes.

In cases where mutually exclusive cells do not exist or are not identified easily, a formal procedure is required. In the next section, the HCA approach to find a deterministic structure in the incidence matrix is formalized.

V. CELL FORMATION USING THE HCA APPROACH

Cell formation using the HCA approach is a 2-step procedure:

1. If A has no deterministic structure, then a best approximative matrix M = [m_{ij}], having a deterministic structure, has to be obtained. The matrix M is called the *recovered incidence matrix.*
2. Among the set of Boolean decompositions of M, the choice is based on which is compatible with the specified set-theoretical relations.

In practice, cases have to be considered in which the hierarchical class model does not represent the data perfectly. In such a case, an approximative model is chosen. For this purpose, the HICLAS algorithm[12] is chosen. It tackles the problem by means of a minimization algorithm, which looks for the best fitting model, given a fixed number of bottom classes (called the rank of the model).[14,15] This minimization algorithm performs the 2-step procedure iteratively. Both steps are explained below.

Step 1: Determine the best approximative deterministic structure of A
HCA assumes that the set-theoretical relations of the underlying binary (incidence) matrix are supposed to be deterministic. If this assumption is violated, HCA finds the incidence matrix M that is the best possible deterministic approximation of the matrix A. The matrix M contains a minimal number of entries that are dissimilar with the recovered entries in A. As a result,

$$A = M + E \text{ or } [a_{ij}] = [m_{ij}] + [e_{ij}]$$

where

$$
eij =
\begin{cases}
0 \text{ if } (a_{ij} = 1 \wedge m_{ij} = 1) \vee (a_{ij} = 0 \wedge m_{ij} = 0) \text{ [similarity]} \\
+1 \text{ if } (a_{ij} = 1 \wedge m_{ij} = 0) \text{ [positive dissimilarity]} \\
-1 \text{ if } (a_{ij} = 0 \wedge m_{ij} = 1) \text{ [negative dissimilarity]}
\end{cases}
$$

with $\quad \Sigma_i \, \Sigma_j \, |\, e_{ij} \,| = \text{minimal}$

E is called the *discrepancy matrix* (or the *error matrix*). De Boeck and Rosenberg[14] state that an enumerative method cannot be used to find the best approximation of A. Instead, an iterative (Boolean) least squares regression procedure is used as a heuristic.[13]

To illustrate the iterative procedure, we introduce the concept of a Boolean decomposition of the (approximated) machine-part incidence matrix M, which consists of 2 matrices S and P, such that:

$$M = S * P'$$
$$\text{(mxn)} \quad \text{(m*r)} \quad \text{(r*n)}$$

with r the Schein[2] rank of M. [The Schein rank is the smallest integer value such that the product of S and P' yields M.]

For example, consider the following decomposition of a (4 × 4) matrix M.

	p1	p2	p3	p4			mB1	mB2			p1	p2	p3	p4	
m1	1	1	1	1		m1	1	1		pB1	1	1	1	1	or
m2	1	1	1	1	=	m2	1	1	*	pB2	0	0	1	1	
m3	0	0	1	1		m3	0	1							
m4	0	0	1	1		m4	0	1							

$$M = S * P'$$
$$\text{(4*4) (4*2) (2*4)}$$

Each column of S and P is called a bundle of machines or parts, respectively. In the matrices the bundles are indicated as mBi (i = 1,2) and pBj (j = 1,2). Often

more than 1 decomposition of M exists. In the illustrated matrix, 2 other decompositions exist:

	S		P'		S		P'
		*		or		*	
	10		1110		01		0011
	01		0011		01		1111
	01				10		
	01				10		

The iterative least squares approach alternates between the rows and the columns of the A matrix to find the best fitting machine and part classes and their hierarchical relations.[14] Given an initial estimate of S or P, the corresponding P or S is estimated. The procedure alternates between the estimation of P and the estimation of S. The matrix product S * P' serves as an estimate of M. The procedure stops when an estimated M is found with a minimal (weighted) number of discrepancies between A and M. Finally, the recovered incidence matrix M is determined.

Before we proceed with step 2, consider the example of mutually exclusive machine cells (Example 2, Matrix 2), the relation between the actual incidence matrix A and its recovered structure M is given by the following equation:

$$[a_{ij}] = [m_{ij}] + [e_{ij}]$$

or

	p1	p2	p3	p4	p5	p6	p7	p8			p1	p2	p3	p4	p5	p6	p7	p8			p1	p2	p3	p4	p5	p6	p7	p8
m1	1	1	1	0	0	0	0	0		m1	1	1	1	0	0	0	0	0		m1	0	0	0	0	0	0	0	0
m2	1	0	1	0	0	0	0	0		m2	1	1	1	0	0	0	0	0		m2	0	-1	0	0	0	0	0	0
m3	1	1	1	0	0	0	0	0		m3	1	1	1	0	0	0	0	0		m3	0	0	0	0	0	0	0	0
m4	1	1	1	0	0	0	0	0		m4	1	1	1	0	0	0	0	0		m4	0	0	0	0	0	0	0	0
m5	0	0	0	1	1	1	0	0	=	m5	0	0	0	1	1	1	0	0	+	m5	0	0	0	0	0	0	0	0
m6	0	0	0	1	1	1	0	0		m6	0	0	0	1	1	1	0	0		m6	0	0	0	0	0	0	0	0
m7	0	0	0	0	1	1	0	0		m7	0	0	0	1	1	1	0	0		m7	0	0	0	-1	0	0	0	0
m8	0	0	0	0	0	0	1	1		m8	0	0	0	0	0	0	1	1		m8	0	0	0	0	0	0	0	0
m9	0	0	0	0	0	0	1	1		m9	0	0	0	0	0	0	1	1		m9	0	0	0	0	0	0	0	0
m10	0	0	0	0	0	0	0	1		m10	0	0	0	0	0	0	1	1		m10	0	0	0	0	0	0	-1	0

The discrepancy matrix contains only 0 entries (similarities) and negative discrepancies (−1 entries). It happens that a part is associated to a given machine class, although it is not processed by all machines which belong to this machine class. For example, p2 is not processed on m2, although m2 belongs to the machine class {m1,m2,m3,m4}. The same is true for p4 and m7, and for p7 and m10. These kind of inefficiencies are represented in the discrepancy matrix as negative dissimilarities. Another type of discrepancy concerns all existing over-

lap between different machine classes or part families. This situation occurs when parts that are not associated to a given machine class are processed on at least one machine belonging to this machine class. The existence of overlap between different machine or part classes is one of the possible indicators of the existence of bottleneck machines or exceptional parts. Overlap corresponds to positive dissimilarities between A and M.

Because the error matrix E in Example 2 contains only negative discrepancies, the machine–cell configuration, which is suggested by the optimal HCA solution (see Matrix 3), contains only inefficiencies, and no bottleneck machines or exceptional parts.

For the formation of machine cells, inefficiencies within machine classes are considered to be less problematical than bottleneck machines or exceptional parts. Therefore, both types of discrepancies should be treated differently in the HCA analyses.

Step 2: The choice of a set-theoretical decomposition

Given the recovered incidence matrix M, a Boolean decomposition has to be found that reflects the 3 set-theoretical relations of equivalence, order, and association. Storms et al.[15] state that the 3 relations are in fact restrictions on the Boolean decomposition. In this step it has to be determined which decomposition represents the set-theoretical structure. De Boeck and Resenberg[14] proved that at least one such decomposition exists.

To test whether a Boolean decomposition is compatible with the 3 relations in M, 2 additional matrices U and V are defined.

$$U = [u_{ij}] \text{ and } V = [v_{ij}]$$

with

$u_{ij} =$ 1 if, in M, machine i is ordered on the same or a higher level than machine j

0 elsewhere,

and

$v_{ij} =$ 1 if, in P, part i is ordered on the same or a higher level than part j

0 elsewhere.

De Boeck and Rosenberg[14] have shown that

$$M = [U*S] * [V*P]'$$
$$\text{(m*n)} \quad \text{(m*r)} \quad \text{(r*n)}$$

Only the matrices S and P that are compatible with the set-theoretical relations expressed in the U and V matrices can be used in the final HCA model.

In the 4×4 example, the extended decomposition looks like

```
    p1 p2 p3 p4        m1 m2 m3 m4      mB1 mB2                              p1 p2 p3 p4
m1  1  1  1        m1  1  1  1  1    m1  1  1                          p    1  1  1  1
m2  1  1  1    =   m2  1  1  1  1  * m2  1  1   *  pB1 1  1  1  1   *   p    1  1  1  1
m3  0  1  1        m3  0  0  1  1    m3  0  1      pB2 0  0  1  1       p    0  0  1  1
m4  0  1  1        m4  0  0  1  1    m4  0  1                          p    0  0  1  1
```

or

$$M = [U * S] * [P' * V']$$

The set-theoretical decomposition of M contains both the product terms [U*S] and [V*P]. The partial order among machine classes and part classes in M is determined by the matrix product [U*S] and [V*P], respectively . In the example, only 1 Boolean decomposition (out of 3) is compatible with the set-theoretical relations in M.

From the matrix U specified in the extended decomposition, it can be derived that machine class {m1,m2} is higher ordered than the bottom machine class {m3,m4}. On the other hand, it is also shown that the part class {p1,p2} is higher ordered than the bottom part class {p3,p4}.

Let S and P be written as a vector of bundles S_i and P_j:

$$S = [S_1 \; S_2 \; ... \; S_i \; ... \; S_n] \text{ and}$$
$$P = [P_1 \; P_2 \; ... \; P_j \; ... \; P_m]$$

with:

    mi = identification of machine i (i=1,...,n)
    pj = identification of part j (j=1,...,m)
    Mi = identification of machine class i
    Pj = identification of part class j
    M   = set of machines
    P   = set of parts.

A bundle-specific machine class BM_i or part class BP_j is defined as

$$BM_i = \{ M_1 \in M \mid \forall k \in M_1 : s_{ki} = 1 \wedge (s_{kj} = 0; i \neq j) \}$$
$$BP_i = \{ P_1 \in P \mid \forall k \in P_1 : P_{ki} = 1 \wedge (P_{kj} = 0; i \neq j) \}$$

{m3,m4} is a bundle-specific machine class. {p1,p2} is a bundle-specific part class.

There is a one-to-one correspondence between machine and part bundles in that all machines from a given machine bundle process all parts from the corresponding part bundle. Bottom machine class {m3,m4} is associated to part class {p3,p4} (p3 and p4 entries in P have value 1 in the second bundle). Bottom part class {p1,p2} is associated to machine class {m1,m2} (m1 and m2 entries in M have value 1 in the first bundle).

Through the order relation associations with lower-ordered classes, additional association relations between classes can be found. Therefore, machine class {m1,m2} is also associated to part class {p3,p4}.

So far, the 2-step procedure has been explained by means of a simple example. It should be stressed that the HCA algorithm considers only alternative models with a maximal number of bottom machine classes and/or bottom part classes. This maximum number is indicated by the number of bundles in S or P (i.e., the rank of the HCA model). In practice, however, the decision maker has to choose among a number of rank-specific optimal HCA solutions. This problem will be dealt with in the next section.

VI. HIERARCHICAL CLASSES ANALYSIS WITH OVERLAP BETWEEN MACHINE CELLS

Often, the formation of mutually exclusive machine cells is prohibited by the existence of bottleneck machines or exceptional parts. By means of an example machine–part incidence matrix (Matrix 4), taken from Sule,[16] we illustrate the HCA approach in determining machine cells and part families concurrently.

HCA solutions can be obtained in different ranks by performing the algorithm described in Section II. The optimal solution in a given rank has a minimal weighted number of discrepancies between A and M. To select the best solution out of a set of optimal solutions in different ranks, one needs a rank-independent goodness-of-fit measure. An adjusted version of the Jaccard similarity coefficient is proposed in which only positive discrepancies are taken into account. Table 4 illustrates the goodness-of-fit statistics for the optimal HCA solutions in ranks 1–5. The adjusted Jaccard similarity coefficient is defined as:

$$\frac{\Sigma_i \Sigma_j \; \delta \, (A_{ij} = 1) \; \delta \, (M_{ij} = 1)}{\Sigma_i \Sigma_j \; \delta(A_{ij} = 1) \; \delta \, (M_{ij} = 1) + \Sigma_i \Sigma_j \; \delta \, (A_{ij} - M_{ij} = 1)}$$

where $\delta \, (p) = 1$ if the proposition p is true; and $\delta \, (p) = 0$ if the proposition p is false.

Note: To determine the optimal solution within a given rank, the weights

MATRIX 4. The machine–part incidence matrix.

	p1	p2	p3	p4	p5	p6	p7	p8	p9	p10	p11	p12	p13	p14	p15	p16	p17	p18	p19	p20
m1	0	0	0	0	1	0	0	1	0	0	1	1	0	0	0	0	0	0	1	1
m2	0	0	0	0	1	1	0	0	0	0	0	1	1	0	0	0	0	0	1	0
m3	1	1	1	0	0	0	0	0	0	0	1	0	0	0	1	1	0	0	0	1
m4	1	1	1	0	0	0	1	0	0	0	0	0	0	0	1	1	0	0	0	1
m5	0	0	0	0	1	0	0	1	0	0	0	1	0	0	0	0	0	0	1	0
m6	0	0	0	1	0	1	0	0	1	0	0	0	0	1	0	0	1	1	0	0
m7	0	0	0	0	1	1	0	0	1	1	0	1	1	1	0	0	1	1	1	0
m8	0	0	0	1	0	0	0	0	1	0	0	0	0	1	0	0	1	1	0	0
m9	1	1	1	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	1
m10	1	1	1	0	1	0	1	1	0	0	1	1	0	0	1	1	0	0	0	0
m11	0	0	0	1	0	0	0	0	1	1	0	0	0	0	0	0	1	1	0	0

$A =$

for a positive discrepancy is set equal to $\frac{2}{3}$, while the weight of a negative discrepancy is set equal to $\frac{1}{3}$. Despite the fact that the weights are chosen somewhat arbitrarily, the optimal HCA solution was found to be robust to moderate adjustments to these weights.

The optimal HCA solution in rank 3 has a better goodness-of-fit value than the optimal HCA solutions in ranks 3, 2, and 1. It is hardly worse than the optimal HCA solution in rank 4. Therefore, the set-theoretical relations in this more parsimonious model is chosen as the best deterministic approximation of the real structure underlying the machine–part incidence matrix.

Table 4. Goodness-of-Fit Statistics of the HCA-Solutions in Different Ranks

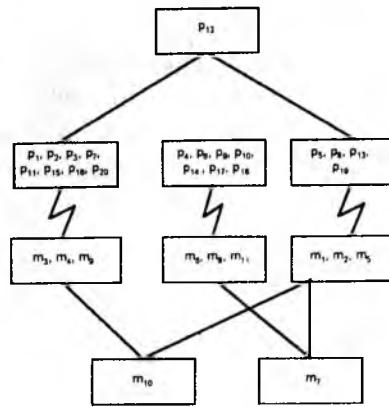| (Optimal solution in) RANK	Number of Similarities		Number of Positive discrepancies	Number of Negative discrepancies	Adjusted Jaccard Similarity Coefficient
	1	0	+ 1	-1	
1	35	129	36	20	0.493
2	57	123	14	26	0.803
3	68	129	3	20	0.958
4	70	131	1	18	0.986
5	71	135	0	14	1.000

FIGURE 2.
The HCA graph of the optimal solution in rank 3.

The HCA graph for the optimal HCA model in rank three is presented in Figure 2.

The optimal HCA solution in rank 3 contains 5 machine classes and 4 part classes. Three of the machine classes—{m3,m4,m9}, {m6,m8,m11} and {m1,m2,m5}—are bottom machine classes . Both other machine classes are higher-ordered machine classes. Three of the part classes—{p1,p2,p3,p7,p11,p15,p16,p20}, {p4,p6,p9,p10,p14,p17,p18} and {p5,p8,p13,p19}—are bottom part classes. The graph shows one higher-ordered part class {p12}. Machines m7 and m10 are bottleneck machines. Part p12 is an exceptional part.

Higher-ordered classes are connected to at least 1 bottom class by a straight line. A bottom part class associated to a bottom machine class is connected in the graph by a zig-zag line. The pattern of zig-zag lines and straight lines in the graph and two rules, to be defined, are used to form the cells. A first rule states that each bottom machine (part) class is associated not only to all parts (machines) in the bottom class to which it is linked, but also to all parts (machines) belonging to any part (machine) class that subsume that class. For example, {m1,m2,m5} is associated to part class {p5,p8,p13,p19} and to part class {p12}.

A second rule states that a machine in a higher-ordered class is used to process all parts associated to the bottom machine class to which the higher ordered one is linked, either directly or indirectly. Similar reasoning can be made for parts.[14] For example, m7 is used to process {p5,p8,p13,p19} and {p4,p6,p9,p10,p14,p17,p18}.

Based on the HCA model, alternative machine cell configurations can be ob-

tained. Making an abstraction of the trivial cell containing all machines, 2 alternatives remain.

A first machine cell configuration can be formed if both machine classes {m7} and {m10} are taken as initial sets of 2 machine cells. To the set {m7}, the machine classes {m3,m4,m9} and {m1,m2,m5} have to be added. To the set {m10}, the machine classes {m6,m8,m11} and {m1,m2,m5} have to be added. Machines m1, m2, and m5 are identified as bottleneck machines. Despite their duplication, p12 remains an exceptional part.

A second machine cell configuration can be formed if the machines in the bottom classes are taken as initial sets of 3 machine cells. In this configuration, machines m7 and m10 have to be duplicated. The resulting machine cells are {m1,m2,m5,m7,m10}, {m3,m4,m9,m10}, and {m6,m8,m7,m11}. Also in this case p12 is an exceptional part.

VII. HIERARCHICAL CLASSES ANALYSIS FEEDING A DECISION SUPPORT SYSTEM

HCA provides a set of feasible and good solutions in terms of goodness-of-fit of a hierarchical model compared to the real situation. Within the set of feasible configurations, a decision-maker (DM) has to choose the alternative that most suits his needs according to either a secondary objective or a subjective evaluation. If the DM is able to specify a (set of) secondary objective(s), a decision-support system, hiding an optimization mechanism, can be worth looking at. In such a case, the DM can include additional information or subjective knowledge not available at the time of performing the HCA analysis. We illustrate this idea with a simple example.

The DM is eager to define independent cells. However, he is limited in budget on the duplication of machines. Therefore, he is confronted with the choice of a discrete number of duplications of machines so that the budget constraint is not violated and the number of intercellular trips is minimized. For this particular case, the DM can include extra data on volumes to measure the intercellular traffic.

Starting from a configuration in which duplicating machines are assigned to the cell with the highest volume for the machine, an 0–1 integer programming optimization problem can be formulated to assign the machines in a most efficient way (i.e., by reducing the intercellular volume), considering the budget. This can be stated as:

$$\max \sum_{i=1}^{m} \sum_{k=1}^{n} V_{ik} X_{ik}$$

subject to

$$\sum_{i=1}^{m} \sum_{k=1}^{n} C_i X_{ik} \leq TB$$

where
  V_{ik} = volume of jobs the ith duplicating machines serves in the assigned cell k
  X_{ik} = 1 if machine i is selected in cell k; 0 otherwise
  C_i = investment cost of machine i
  TB = total available budget

In operations research terminology, this optimization is known as the *knapsack problem,* but in a decision-support system the analyst does not have to be aware of which problem it is or which solution method is being used.

VIII. CONCLUSION

The new Hierarchical Classes Analysis (HCA) approach is introduced as a de-cision-supporting instrument for the formation of machine cells.

The HCA approach consists of an iterative procedure to find a set-theoret-ical decomposition of a recovered machine–part incidence matrix. The optimal HCA solution can be graphically presented in an HCA graph, which consists of a simultaneous hierarchical representation of machine and part classes.

The HCA graph is very helpful in identifying the machines or parts that ob-struct the creation of mutually exclusive machine cells. In addition, it has been shown that alternative machine cell configurations can easily be derived from the HCA graph.

As opposed to cluster analysis (CA) approaches, the HCA approach is con-cerned with finding a hierarchical representation of the machine (class) and part (class) structure, simultaneously.

REFERENCES
1. **Kusiak, A. and Chow, W. S.,** Decomposition of manufacturing systems, *IEEE J. Robot. A.,* 4(5), 457–471, 1988.
2. **Burbridge, J. L.,** *Production Flow Analysis,* Oxford: Clarendon Press, 1989.
3. **Kusiak, A.,** *Intelligent Manufacturing Systems,* Englewood Cliffs, NJ: Prentice Hall, 1990.

4. **Logendran, R.,** Impact of sequence of operations and layout of cells in cellular manufacturing, *Int. J. Prod.,* 29(2), 375–390, 1991.
5. **Nagi, R., Harhalakis G., and Proth, J.-M.,** Multiple routings and capacity considerations in group technology applications, *Int. J. Prod.,* 28(12), 2234–2257, 1990.
6. **Taboun, S. M., Sankaran, S., and Bohle, S.,** Comparison and evaluation of similarity measures in group technology, *Comput. Indust. E.,* 20(3), 343–353, 1991.
7. **Miltenburg, J. and Zhang, W.,** A comparative evaluation of nine well-known algorithms for solving the cell formation problem in group technology, *J. Operat. Man.,* 10(1), 44–72, 1991.
8. **King, J. R. and Nakornchai, V.,** Machine part group formation in group technology—Review and extension, *Int. J. Prod.,* 20(2), 117–133, 1982.
9. **Eckes, T.,** Features of situations: A two-mode clustering study of situation prototypes, *Personality and Social Psychology,* 21(4), 366–374, 1995.
10. **Arabie, P. and Hubert, L. J.,** The bond energy algorithm revisited, *IEEE Transactions on Systems, Man and Cybernetics,* 20, 268–274, 1990.
11. **McCormick, W. T., Schweitzer, P .J., and White, T. W.,** Problem decomposition and data reorganization by a clustering technique, *Operations Research,* (20(5, 993-1009, 1972.
12. **Eckes, T. and Orlik, P.,** An error variance approach to two-mode hierarchical clustering, *J. Classification,* 10, 51–74, 1993.
13. **Mickey, M. R., Mundle, P., and Engelman, L.,** Boolean factor analysis, in *BMDP Statistical Software,* Dixon, W. J., Ed., Berkeley, CA: University of California Press, 538–545, 1983.
14. **De Boeck, P. and Rosenberg, S.,** Hierarchical classes: Model and data analysis, *Psychometrika,* 53(3), 361–381, 1988.
15. **Storms, G., Van Mechelen, I., and De Boeck, P.,** Structural analysis of the intension and extension of semantic concepts, *Eur. J. Cognitive Psychology,* 6(1), 43–75, 1994.
16. **Sule, D. R.,** *Manufacturing Facilities Location, Planning and Design,* Boston: PWS Publishing 722, 1994.