SEQUENCES CLASSIFICATION BASED ON GROUP TECHNOLOGY FOR FLEXIBLE MANUFACTURING CELL DESIGN

Akbib Maha[1], Amrani Ghacham Abdellatif[2], Sedqui Abdelfettah[3]

[1]*Phd student in logistics, Innovative Technologies Laboratory, ENSAT, Tangier, Morocco,* ***maha.akbib@gmail.com***
[2]*Professor of electronic engineering, Physics Department, ENSAT, Tangier, Morocco,* ***ab_amrani@yahoo.fr***
[3]*Professor of the industrial and logistic engineering, Physics Department, ENSAT, Tangier, Morocco,*
abdelfettah.sedqui@gmail.com

Abstract

Flexible cell formation is based on Group Technology. Group Technology rests on the exploitation of resemblances between products or processes, which makes the identification of products' families and machines' cells easier. We propose a new approach based on the language theory for product family grouping according to their manufacturing sequences. This approach uses linear sequences of the manufacturing products which are assimilated to the words of a language. We have chosen the Levenhstein distance for sequence classification. We are going to compare our method to Dice-Czekanowski and Jaccard's methods and apply the vectorial correlation coefficient as a comparison tool between two hierarchical classifications.

Keywords: *manufacturing sequences, language theory, hierarchical classification, Group Technology.*

---***---

1. INTRODUCTION

Because of the diverse needs of the customers and the proliferation of the product, the manufacturers are faced to difficulties treating the frequent change of the conception and the recurring variations of the processes, which increase the complexity of the product and the structures of production [1] [2]. The development of multiple products based on product families sharing a common platform was well recognized as a successful approach in many industries [3].

The Group Technology (GT) is an organization principle which can be applied to all the domains of an industrial company. This principle rests on the combination of identical or similar components to define groups, classes or families by using a system of classification [4]. The purpose is to decompose the industrial system into sub-systems which are easier to control.

The main stage in the design of industrial systems is to group products into families and machines into cells in which one or several product families are made [5]. A family of product is a collection of the products which are similar as regards their geometrical forms, their dimensions and/or the presentation of similar stages in the manufacturing process. This work considers the last criterion. The number of studies dedicated to the determination of product families by taking into account the order of the operations is limited [6] [4] [7] [8] [9]. This is due to the difficulty of determining a resemblance measure verifying some severe criteria [10] as to know the non-coincidence of operation, the order of operation, the permutation of operation and the number of operation. With the aim of designing manufacturing systems, we propose a new approach based on the theory of language for the determination of sequences families. It takes into account all the previous approaches based on the typological analysis and more particularly the hierarchical classification.

Thus, in the 2nd part, we propose a measure of resemblance for the sequences classification, suiting in the industrial case and answering the above-mentioned criteria [4] [10]. In the 3rd part, the results of the method and the interpretations are presented. And as regards the 4th and last part, it will contain the comparison between our method and the methods of Jaccard [11] and Dice-Czekanowski [12].

2. THE PROPOSED MEASURE OF RESEMBLANCE

Several approaches were developed to identify product families and the associated machine cells. These approaches can be classified according to two groups [13]:

- Approaches based on the characteristics of the product,
- Approaches based on the methods of fabrication.

These approaches use measures of resemblance applied in search of the product families and the machine cells. Two components are necessary [14]:

- A metric (the distance, the index of similarity…),
- A classification algorithm.

2.1 The Language Theory Approach

For the grouping of products in families, it is reasonable to develop partitions by making comparisons between the ranges of manufacturing [15]. To this end, the treated sequences are linear sequences which are similar as for their manufacturing process. The measure of resemblance proposed for the classification of these sequences has to verify the following criteria [4] [10]:

Table -1: The four criterion

Physical criterion	Property
C1:Non-coincidence	The more the non-coincidence increases, the more the index of similarity grows
C2:Order of the operations	The order of the operations is faithfully translated
C3:Permutation	The permutation of two consecutive phases has less incidence on the index than any other permutation
C4:Number of operations by sequence	The difference of K operations between two short sequences must have more influence on the index than for two long sequences

To solve this problem, we notice the analogy existing between the words of a language and linear sequences of products.

Accordingly, we propose the assimilation of the product sequences to the words of a language which is defined by using an alphabet (a set of machine operations).

We choose the balanced Levenhstein distance, so called the distance of edition [16] which verifies the above criteria, see the table 2 at the end of the chapter 2.

2.2 Calculation of the Distance and the Application to the Manufacturing Sequences

The following definitions will be used:

- A linear sequence is defined as a sequence in which the operations are executed in the sequential order.
- An empty operation is an operation which corresponds to a no letter in the theory of language.
- An empty sequence corresponds to a no word in the theory of language.

Three operations are used for the calculation of the edition distance between the sequences S1 and S2:

- The substitution of an operation b of the sequence S2 by an operation a of the sequence S1, noted $a \rightarrow b$ and the cost of which is c (a, b),
- The insertion of an operation in a manufacturing sequence, $\mu \rightarrow a$, the cost of which is c (a,μ) with μ the empty operation,
- The deletion of an operation of a manufacturing sequence, $a \rightarrow \mu$, the cost of which is c (a, μ), μ the empty operation.

Example

The sequence S1 = ab can be changed in sequence S2 = bac by the following set of elementary operations:

$$(\mu \rightarrow c), (b \rightarrow \mu), (c \rightarrow b), (\mu \rightarrow c)$$

Which corresponds to the stages of transformation: ab, cab, ca, ba, bac. The cost of this transformation which is not optimal is:

$$c(\mu, c)+c(b, \mu)+c(c, b)+c(\mu, c)$$

The distance d (S1, S2) is the cheapest order of the elementary transformation necessary to change S1 in S2. To calculate d (S1, S2) and avoid the combinatorial explosion, we have to consider the natural constraint: d (a, b) = c (a, b) which consists in replacing an operation a by the operation b implying the triangular disparity: $c(a,b)+c(b,e) \geq c(a,e)$

This condition allows the determination of the minimal cost transformation in a finite set.

The transformation of the succession corresponds at the minimum of necessary steps to change S1 in S2, using the three operations described below. As an example, the final result is represented in the following way:

```
S1= a1   a2    a3 ... an     μ
     |    |     |      |     |
S2= μ    b1    b2 ...bn-1   bn
```

The operations connected by a line indicate a substitution. The operations connected with the empty operation μ are insertions or deletions. It is as well compulsory as two lines do not have to cross inside a sequence and do not particularly have to begin of (or to end on) the same operation.

S1 is thus transformed into S2 by:

$$(a1 \rightarrow \mu), (a2 \rightarrow b1), ..., (an \rightarrow bn\text{-}1), (\mu \rightarrow bn)$$

2.2.1 Property [17]:

The cost of a minimal cost transformation is equal to the cost of a minimal cost succession.

Thus, the particular shape of successions allows the introduction of a recursion relation with the minimal cost transformation.

$$S1 = a_1 \ldots a_n$$
$$S2 = b_1 \ldots b_m$$
$$S1(i) = a_1 \ldots a_i$$
$$D(i, j) = d(S1(i), S2(j))$$
$$D(i, j) = \min \begin{cases} D(i-1, j-1) + c(a_i, b_j) \\ D(i-1, j) + c(a_i, \mu) \\ D(i, j-1) + c(\mu, b_j) \end{cases}$$

2.2.2 Algorithm for the Calculation of the Distance:

The additivity of the criterion of cost as for the elementary cost assures that a method of dynamic programming can be then used. It is translated by the following algorithm [16]:

1) D(0, 0) = 0
2) for i:=1 to n do
   D(i, 0):= D(i-1, 0) + c(ai , µ)
3) for j:= 1 to m do
   D(0, j):= D(0, j-1) + c(µ, bj)
4) for i:=1 to n do
   for j:=1 to m do

$$D(i, j) = \min \begin{cases} D(i-1, j-1) + c(a_i, b_j) \\ D(i-1, j) + c(a_i, \mu) \\ D(i, j-1) + c(\mu, b_j) \end{cases}$$

5) d(S1, S2) = D(n, m).

2.2.3 Example of Execution:

Let O = {a, b, c, d} be the set of operations to be executed on machines A, B, C and D. And let two manufacturing sequences S1 = a a b a c d and S2 = a b d, c(x, y) = 1 for every $x, y \in O \cup \{\mu\}$ and $x \neq y$, c(x, x) = 0 for every $x \in O$.
The execution step by step of the algorithm of [16] gives the table 2, whose last element represents the result.

Table -2: Distance between sequences abd et aabacd

	µ	a	a	b	a	C	d
µ	0	1	2	3	4	5	6
a	1	0	1	2	3	4	5
b	2	1	1	1	2	3	4
d	3	2	2	2	2	3	**3**

Sequence corresponding to a distance of 3 between S1 and S2:

a a b a c d
| | | | | |
a µ b µ µ d

Remark:

The links correspond to a minimal path by the matrix of distance from the first element to the last one, see table 2.
This distance verifies exactly the above mentioned criteria, as indicated in the table 3:

Table-3: Verification of the criteria by the proposed distance

Criteria	sequences	distance
non-coincidence	S1= abc S2= afc S1= abc S2= efg	d(S1, S2) = 1 d(S1, S2) = 3
permutation	S1= abc S2= acb S1= acb S2= cab	d(S1, S2) = 2 d(S1, S2) = 2
Operations order	S1= abc S2= abc S1= abc S2= bac	d(S1, S2) = 0 d(S1, S2) = 2
Number of the operations in a sequence	S1= abc S2= a S1= abc S2= ab	d(S1, S2) = 2 d(S1, S2) = 1

3. RESULTS, VALIDATION AND INTERPRETATION

3.1. Basic Data

We proposed the consideration of linear sequences as being similar to the words of a language. As a consequence, a light modification in their representation is necessary with regard to the previous work. To estimate the results of the method, we apply the proposed distance to the sequences described in the table 4. The sequences are noted Si, with i=1… 31.

Table -4: Examples of linear sequences

S1	TEHKSF	S17	ADPEGKJNRQ
S2	TCHKF	S18	AEKLIHFN
S3	TDCHKF	S19	ABKMGIJN
S4	TCGHK	S20	ABHKL
S5	TDCHKEF	S21	AEHKSMIQFN
S6	TCIHKMEF	S22	ACHKLF
S7	TCHKMUQFN	S23	ACHKF
S8	TCHKMUQFN	S24	AEGHKF
S9	TDCHKLEF	S25	CHKF
S10	TDCHKLEF	S26	ABHKMIQFN
S11	ABPEHIKM	S27	ADHKLEMIQFN
S12	ACPEHKLMOJRQSN	S28	ADHKLEMIQFN
S13	ACPEHK	S29	ABK
S14	ADPEHKLSMOQN	S30	ABHKLMIFN
S15	ACPEHKMORQN	S31	ACHKEF
S16	ADPEGKJN		

The distance of edition was applied to these sequences with the following conditions:

- c(a, a) = 0 for every operation,
- c(a,b) = 1 with a≠b (in this case the operations of transformation : insertion, substitution and deletion are applied) for every operation b ∈ O ∪ { μ }.

We obtain a matrix of distances which we are going to use for the determination of the classification tree. By putting Si=i and by applying the algorithm of ascending hierarchical classification which takes as criterion of aggregation the average link, we arrive at the following dendrogramme (figure 1):

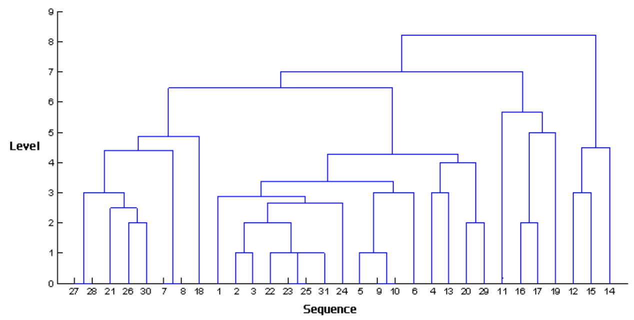

Fig -1: Hierarchical classification P1 of sequences with Levenhstein distance

3.2. The Sequences Families

The determination of sequences families requires the cut of the classification tree in the appropriate threshold.

The determination of the threshold α is easy. When we choose α = 6, we obtain four families of sequences described in the table 5. Tables 5.a, 5.b, 5.c and 5.d specify elements and profile of every family;

Table -5: families obtained with α =6

Family n° 1: number of sequences 16, 51.6 % of the total number.	**F1={ 1, 2, 3, 4, 5, 6, 9, 10, 13, 20, 22, 23, 24, 25, 29, 31 }**
Family n° 2: number of sequences 4, 12.9% of the total number.	**F2={ 11, 16, 17, 19 }**
Family n° 3: number of sequences 3, 9.6% of the total number.	**F3={ 12, 14, 15}**
Family n° 4: number of sequences 8, 25.8% of the total number.	**F4={ 7, 8, 18, 21, 26, 27, 28, 30}**

Table -5.a: Elements and profile of the family F1

sequences		**sequences**	
S1	TEHKSF	S13	ACPEHK
S2	TCHKF	S20	ABHKL
S3	TDCHKF	S22	ACHKLF
S4	TCGHK	S23	ACHKF
S5	TDCHKEF	S24	AEGHKF
S6	TCIHKMEF	S25	ACHKF
S9	TDCHKLEF	S29	ABK
S10	TDCHKLEF	S31	ACHKEF
Profile of family F1: Short sequences characterized by the occurrence of successions CHKF and AHK			

Table -5.b : Elements and profile of the family F2

sequences		**sequences**	
S11	ABPEHIKM	S17	ADPEGKJNRQ
S16	ADPEGKJN	S19	ABKMGIJN
Profile of family F2 : Long sequences characterized by the occurrence of the succession APEK			

Table -5.c : Elements and profile of the family F3

Sequences	
S12	ACPEHKLMOJRQSN
S14	ADPEHKLSMOQN
S15	ACPEHKMORQN
Profile of family F3 : Long sequences characterized by the occurrence of the succession APEHKMOQN	

Table -5.d : Elements and profile of the family F4

sequences		**sequences**	
S7	TCHKMUQFN	S26	ABHKMIQFN
S8	TCHKMUQFN	S27	ADHKLEMIQFN
S18	AEKLIHFN	S28	ADHKLEMIQFN
S21	AEHKSMIQFN	S30	ABHKLMIFN
Profile of family F4: Short sequences characterized by the occurrence of successions HKMFN and AKIFN			

4. COMPARISON

We are going to compare our hierarchical classification, obtained by the use of the distance of Levenhstein, to those obtained by using:

- Dice-Czekanowski distance [12], (figure 2)
- Jaccard distance [11], (figure 3)

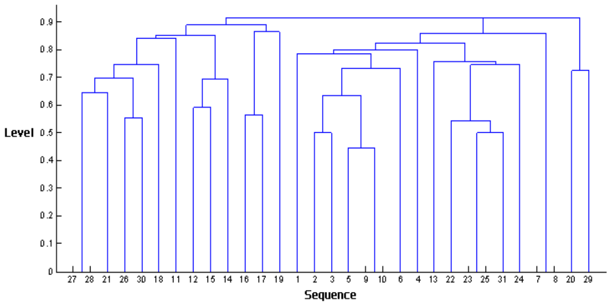

Fig -2: Hierarchical classification P2 of sequences with Jaccard distance

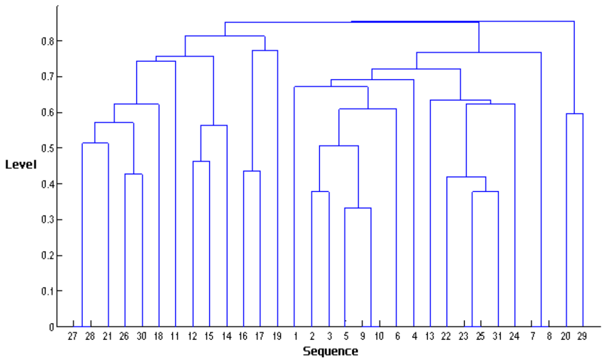

Fig -3: Hierarchical classification P3 of sequences with Dice-Czekanowski distance

In the same way, we cut these two hierarchical classifications. We choose α = 0.8 for figure 3 and α = 0.87 for figure 2. Then we obtain four sequences families described in tables 6 and 7.

Table -6: families obtained with α =0,87 and Jaccard distance

Family n° 1: number of sequences 16, 51.6 % of the total number.	**F1={ 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 13, 22, 23, 24, 25, 31 }**
Family n° 2: number of sequences 3, 9.6 % of the total number.	**F2={ 16, 17, 19 }**
Family n° 3: number of sequences 10, 32.2 % of the total number.	**F3={ 11, 12, 14, 15, 18, 21, 26, 27, 28, 30 }**
Family n° 4: number of sequences 2, 6.4 % of the total number.	**F4={ 20, 29 }**

Table -7: families obtained with α =0,8 and Dice-Czekanowski distance

Family n° 1: number of sequences 16, 51.6 % of the total number.	**F1={ 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 13, 22, 23, 24, 25, 31 }**
Family n° 2: number of sequences 3, 9.6 % of the total number.	**F2={ 16, 17, 19 }**
Family n° 3: number of sequences 10, 32.2 % of the total number.	**F3={ 11, 12, 14, 15, 18, 21, 26, 27, 28, 30 }**
Family n° 4: number of sequences 2, 6.4 % of the total number.	**F4={ 20, 29 }**

To compare the 3 classifications P1, P2 and P3, we are going to use the index of vectorial correlation RV [18]. This index is formulated as follows [19]:

$$RV\ (P1, P2) = \frac{\sum_{i,j}(C_{ij}^1)(C_{ij}^2)}{\sqrt{\sum_{i,j}\left(C_{ij}^1\right)^2 \sum_{i,j}(C_{ij}^2)^2}}$$

With : Ck is the relational table associated with Pk, whose general term C_{ij}^k is defined by :

$$C_{ij}^k = \begin{cases} 1 & \textit{if the sequences i and j are in the same class of partiti} \\ 0 & \textit{otherwise} \end{cases}$$

We find that: RV (P1,P2) = 0.7259
RV (P1,P3) = 0.7259
RV (P2,P3) = 1

Thus, as regards classifications P2 (distance of Jaccard) and P3 (distance of Dice-Czekanowski), they are identical, The only difference is that the distance of Dice-Czekanowski allows a better spreading and a better discrimination. It can be explained by the fact that these two distances can be put under the general shape [12]:

$$D = \sqrt{\frac{N}{\propto P + N}}$$

With: respectively $\propto = 2$ *and* $\propto = 1$ for Dice-Czekanowski and Jaccard.
P is the copresence
N is the non-coincidence

The coefficient RV between P1, P2 and P1, P3 is big enough to judge that the hierarchical classifications are nearby.

CONCLUSIONS

The theory of language approach opens a very interesting direction for the determination of families of product. On one hand, it allows the resolution of the representation problem of operations which are repeated several times in an order. Thus the method eliminates the distribution of error in the calculation of distance and, accordingly, in the results of classification process. On the other hand, this approach allowed the adaptation of the Levenhstein balanced distance in the case of the linear manufacturing sequences. The distance verifies the criteria underlined by [4] and [10] for the determination of product family. The distance:

- Applies to the sequences of different lengths,
- Takes into account permutations of operations,
- Takes into account the order of operations,
- Does not take into account the co-absences, which can increase the similarity of the sequences.

This is due to the conception of the distance which is based on three fundamental operations: substitution, insertion and deletion of elements. Besides, the distance uses directly raw data without transcoding.

For the determination of the sequences families, the frankness of the threshold determination, allows the introduction of new subjective criteria for the characterization of product family.

This approach opens new perspectives for the representation of sequences families by means of finite state automaton. Then the decision-making for the classification of new products becomes simpler and makes the search for the main sequences more determined.

REFERENCES

[1]. E. Westkämper, Th. Schmidt, H.-H. Wiendahl, "Production planning and control with learning technologies: Simulation and optimization of complex production processes ", in: G. Leondes (Ed.), Knowledge-based Systems, vol. 3, Academic Press, New York, 2000.
[2]. J.Jiao, L. Zhang, S. Pokharel, Z. He, "Identifying generic routings for product families based on text mining and tree matching", Decision Support Systems 43, pp. 866–883, 2007.
[3]. S.W. Sanderson, M. Uzumeri, " Managing Product Families", McGraw-Hill Management and Organization Series, Singapore, 1997.
[4]. A. Nadif, M. Constantini, B.Mutel "Mesures de ressemblance de gammes de fabrication" APII, 1985-19. pp. 455-470.
[5]. I. Al-Qattan, "Designing Flexible Manufacturing Cells using a Branch and Bound Method" Int. J. Prod. Res., 1990, Vol. 28, N° 2, p.325-336
[6]. J. Louati, "Similitudes et formes fortes en Technologie de Groupe: Application aux méthodes industrielles dans les P.M.E.(Petites et Moyennes Entreprises)", Thèse de Docteur-Ingénieur, ENSAM, 1986.
[7]. G. Shiko,"Application de la Technologie de Groupe au Bureau des Méthodes", Thèse de Docteur-Ingénieur, ENSAM Paris, 1983.
[8]. A. Talbi,"Contribution à l'étude de la reconnaissance de familles de pièces en fabrication mécanique", Thèse de Docteur, Université de Metz, Decembre 1989.
[9]. A. Sedqui, "Nouvelle approche pour la classification des gammes de production", Thèse de doctorat de l'Institut National des Sciences Appliquées de Lyon, 1995.
[10]. B. Mutel, A. Bryand:"Une nouvelle approache for la reconnaissance de familles de produits", CIM Bordeaux, 1990.
[11]. McAuley J. "Machine grouping for efficient production". Prod Eng, 1972.
[12]. A. Nadif,"Contribution à la classification automatique des données de production en Technologie de Groupe", Thèse de Docteur, Université de Metz, Septembre 1987.
[13]. J.C. Wei and G.M. Kern, "Commonality Analysis: A linear cell clustering algorithm for Group Technology" Int. J. Prod. Res., 1989,Vol. 27, N° 12, pp. 2053-2062.
[14]. C.T. Mosier, "An experiment investigating the application of clustering procedures similarity coefficients to the G.T. machine cell formation problem", Int.J.Prod.Res.,1989,Vol. 27, N° 10, p. 1811-1835.
[15]. J. Minot, B. Mutel, Y. Lemoine, "Implantation assistée par ordinateur de la technologie de groupe" Congrès AFCET Productique et Robotique, Besançon, 1983, pp. 83-96.
[16]. R.A. Wagner, M.J. Fisher "The string to string correction problem" JACM, Vol. 21, N° 1, pp. 168-173, January 1974.
[17]. L. Miclet, "Méthodes structurelles pour la reconnaissance de formes" Edition Eyrolles, Paris, 1984.
[18]. P. Robert, Y. Escoufier, "A Unifying Tool for Linear Multivariate Statistical Methods: the RV-coefficient". Appl. Statist., vol. 25**,** 257-265, 1976.
[19]. G. YOUNESS, "Contributions à une méthodologie de comparaison de partitions", thèse de doctorat de l'université paris 6, 2004.