RESEARCH ARTICLE OPEN ACCESS

Automatic Recognition of Isolated And Interacting Manufacturing Features In Milling Process

Abdelilah El Mesbahi*, Oussama Jaider *, Ahmed Rechia*
*(Research team in Engineering, Innovation, and Management of Industrial Systems, Abdelmalek Essaadi University, Faculty of Sciences and Technics of Tangier, Morocco)

ABSTRACT
Manufacturing features play an important role between design information and manufacturing activities. Recently, various efforts have been concentrated in development of automatic feature recognition systems. However, only limited number of features could be recognized, intersecting features were generally not involved. This paper presents a simple system, in which manufacturing features are easily detected using a Chain of Faces and Base of Faces (CF-BF) graph. A feature is modeled by a series/parallel association of opened Chain of Faces (OCF) or Closed chain of Faces (CCF) that rest on a Base Face (BF). The feature is considered Perfect Manufacturing Feature (PMF) if all Faces that participate in constitution of OCF/CCF are blank faces, else it is an Imperfect Manufacturing Feature (IMF). In order to establish news Virtual Faces to satisfy this necessaries condition, a judicious analysis of orientation of frontier faces that rest on BF is performed. The technique was tested on several parts taken from literature and the results were satisfying.
Keywords – CAD/CAPP/CAM, Automatic Feature Recognition, STEP, Milling Process

I. INTRODUCTION

Automatic Feature recognition has allowed a crucial interest in recent years because it is considered as a key for linking Computer Aided Design (CAD) activities and Manufacturing Aided Design (CAM) activities, to achieve a complete Computer Aided Process planning (CAPP). According to Chee and Kher [1], the approaches for building the CAD/CAM interface can be classified into two main categories: design by features and feature recognition. This second category can be subdivided into: interactive recognition of features and automatic feature recognition. It was found that feature interaction classifications available in the literature are strongly oriented towards the feature recognition approach and are mainly inappropriate to design-by-features systems, because automatic recognition of features has the advantage that the designer does not need to have a deep manufacturing knowledge and he has more time to study the form of the desired part and its functional aspects [2]. Furthermore the creativity of the designer for building innovative component is not affected [3].

Feature traduces different meaning in different contexts depending on the specific domain [4]. For example, in design, a feature refers to a web or a notch section, while in manufacturing, it refers to slots, holes, and pockets. So there are numerous definitions available in the literature for the term “feature”: “regions of a part having some machining significance” [5], “solids removable by operations typically performed in a 3-axis machining center” [6], and “elements used in generating, analyzing, or evaluating design” [7].

In the area of manufacturing features recognition, many techniques have been developed and implemented such as Attribute Adjacency Graph (AAG) [5], volume decomposition [8], hybrid approaches, [9] syntactic pattern recognition [10] and other methods [11], [12]. Most of approaches listed above are developed for a specific geometry of parts such as rotational or prismatic. Furthermore, only few systems have the ability to identify features interactions and give some alternative interpretations of interacting features. For the purpose, this paper proposes a simple methodology for identification of isolated and interacting manufacturing features. Features detection is based on the concavity of edges/faces proposed by Kyprianou [13] and extrapolated later by Xu and Handuja [14]. The method has the ability to give alternative interpretations of identified-interacting features.

II. PREVIOUS WORKS

There are two main solid modeling representations, Boundary representation (B-rep) and constructive solid geometry (CSG). The B-rep of a solid model contains information about faces, edges, and vertices of a surface model including topological information that defines the relationship between faces, edges and vertices [15]. To specify the material side of the object, the normal of B-rep-surfaces is conventionally defined to point toward the exterior of the object. There are a number of features extraction systems proposed in literature. Here, we focus on the work that is more closely related to our approach principally that is related to recognition of interacting features.

Joshi and Chang [5] developed a system based on sub-graph isomorphism to match feature patterns to patterns in the topology of polyhedral parts. The system designed by Attributed Adjacency Graph (AAG) is built from the information contained in B-rep representation. Nodes in graph represent faces of the part, arcs denote edges charred by two adjacent faces and arc attribute represents the concavity/convexity of edge. This approach covers six types of features, but just handles feature interaction for only two of the interactions possible for the six types.

Gao and Shah [16] proposed a feature recognition system based in a minimal condition sub-graph (MCSG) used as a feature hint. The system is capable of recognizing both isolated and interacting features in a uniform way. Hints are defined by an Extended Attributed Adjacency Graph (EAAG), generated by graph decomposition and completed by adding virtual links, corresponding to entities lost by interactions.

Samarghandy and Li [3] have presented an algorithm for the construction of feature volumes using B-rep of a solid object. Faces adjacent to the feature faces are intersected to create new edges that can be used to create new construction faces until construction of the total feature volume. Following the authors, the method only describes the handling of intersections where only one intersecting curve occurs, but, it is unable to handle situations where two or more intersecting curves are produced, also the validity of the feature is not performed during the feature recognition process.

Both methods using intersection of adjacent surfaces and hints are principally based in the possible extrapolation (extension) of adjacent surfaces, to give the required edge or hint for recuperation of all features from interacting features. However, we think that we cannot always ensure the extrapolation if the adjacent surfaces are complex. Furthermore, in general, adjacent surfaces are not of planar types and so, there is no guaranty that these surfaces can intersect the feature face to generate a hint or give the required edge for construction of the feature.

Zulkifli and Meeran [17] used a Kohonen self-organizing feature map (SOFM) neural network for decomposing interacting features. Decomposition process utilizing Boolean operations intersects the resultant area with the maximal rectangular regions (MRR) to generate regions that represent primitive features, referred to, as primitive regions. These primitive regions are then subtracted from the resultant area. Any remaining region is further decomposed into primitive regions, using a second stage of the SOFM and decomposition process. This method is not general for all interacting feature and it is described for only prismatic parts, but it permits to avoid the combinatorial explosion like those in many other systems.

III. FRAMEWORK

3.1 CAD INPUT FILES OF PART DESIGN

Methods for access to topological and geometrical information related to the part from CAD modelers can be classified as internal and external. Internal approaches comprehend use of API (Application Protocol Interface) of the software by which the part was designed. On contrary, external approaches CAD model of the part is exported from software by which it was designed in a neutral data format (STEP, IGES, ACIS, etc.) [17]. Due to the large variety of CAD systems in the market, data exchange between different CAD systems has become indispensable and consequently, neutral data formats (STEP, IGES…,) constitute a common language for interfacing among these different CAD platforms. Among all neutral data formats, recently, STEP has allowed many attentions from the others due to the capacity of describing part's geometry, topology, and tolerances, relations with other parts, various attributes and contingence to appropriate assembly. In the case of CAD/CAM, this format provides detailed information needed to manufacture the required part, including the materials, part geometry, dimensions and tolerances. STEP representation is based on an ingenious B-Rep representation which incorporates the topological information into the geometric information. But in STEP format, geometric entities description is more explicit of that in B-rep. STEP file format is the unique neutral file format that uses the object oriented database structure to map the relationship within the file data structure [18]. The structured information within the STEP file can be explained through the part represented in figure 2a, and the reorganized-excerpt of it STEP file is described below in figure 1. The geometrical and topological entities of part STEP file are designed each one by a specific Keywords PLANE, LINE, CARTESIEN_POINT and ORIENTED_EDGE, VERTEX_POINT respectively. Each entity is indexed by a pointer that makes it easy direct access. In figure 2b, we have designed Face, Edge

```
#51=CLOSED_SHELL('Closed Shell',(#91,#122,#153,#184,#215,#237,#251,#265)) ;
  #184=ADVANCED_FACE('Corps principal',(#183),#158,.T.) ;
    #183=FACE_OUTER_BOUND('',#178,.T.) ;
       #178=EDGE_LOOP('',(#179,#180,#181,#182)) ;
             #179=ORIENTED_EDGE('',*,*,#141,.F.) ;
                    #141=EDGE_CURVE('',#133,#140,#138,.T.) ;
                        #133=VERTEX_POINT('',#132) ;
                              #132=CARTESIAN_POINT('Vertex',(0.,30.,15.)) ;
                        #140=VERTEX_POINT('',#139) ;
                              #139=CARTESIAN_POINT('Vertex',(55.,30.,15.)) ;
                        #138=LINE('Line',#135,#137) ;
                              #135=CARTESIAN_POINT('Line Origine',(27.5,30.,15.)) ;
                              #137=VECTOR('Line Direction',#136,1.) ;
                                    #136=DIRECTION('Vector Direction',(1.,0.,0.)) ;

    #158=PLANE('Plane',#157) ;
       #157=AXIS2_PLACEMENT_3D('Plane Axis2P3D',#154,#155,#156) ;
            #154=CARTESIAN_POINT('Axis2P3D Location',(0.,30.,15.)) ;
            #155=DIRECTION('Axis2P3D Direction',(0.,0.,-1.)) ;
            #156=DIRECTION('Axis2P3D XDirection',(0.,1.,0.)) ;
```

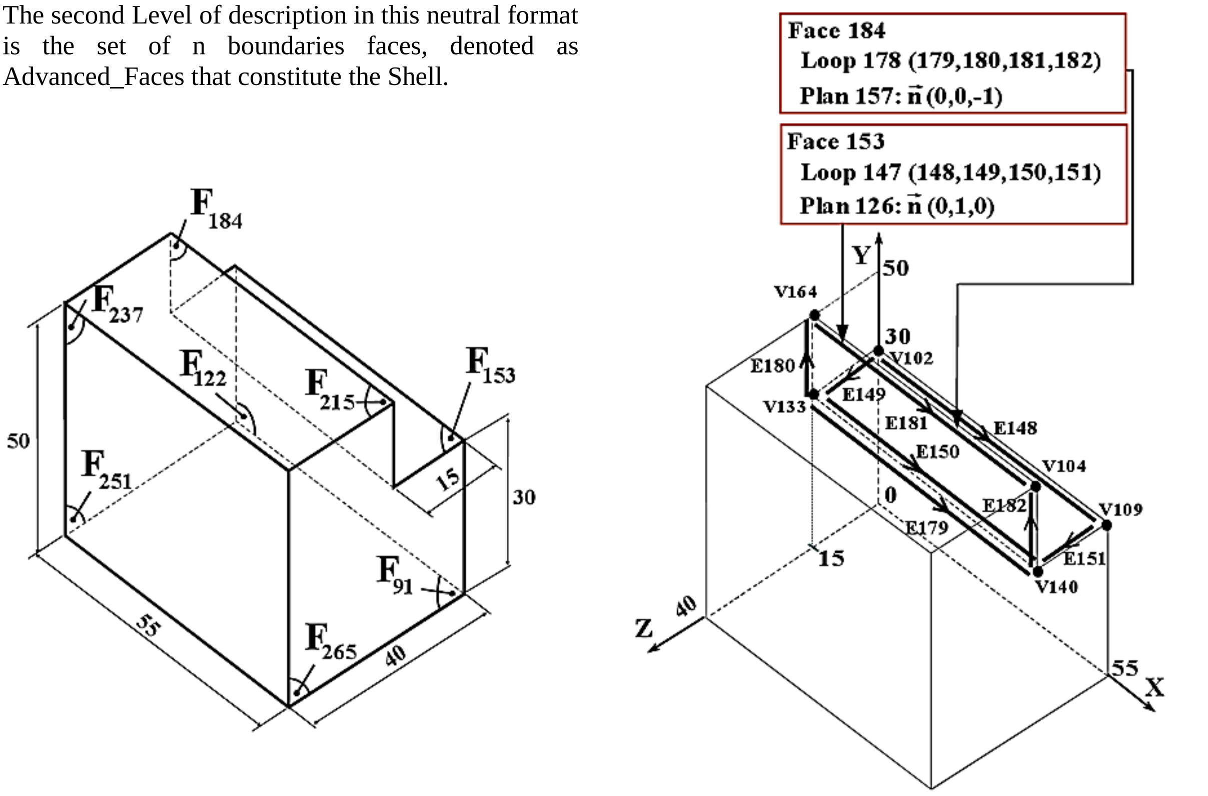
Figure 1: An organized excerpt of STEP file of the example part of figure 2

and Vertex by F, E, and V respectively. There index are taken to be the same of its pointers. The first level entity in STEP is the Shell. A shell is an enclosed volume delimited by joining faces along edges. This domain is connected, oriented, non-self-intersecting surfaces.

The part of figure 2 is constituted by only one shell:Shell:#51=CLOSED_SHELL('Closed Shell',(#91,#122,#153,#184,#215,#237,#251, #265)) where #51 is the pointer of this Shell. and #91, #122, #153,#184, #215, #237, #251, #265 are the pointers of it boundaries faces.

The second Level of description in this neutral format is the set of n boundaries faces, denoted as Advanced_Faces that constitute the Shell.

An Advanced_Face in STEP is a topological entity that is defined in terms of geometric and topological information. For example, face of address #184 designed in figure 2a by F184 is giving by the following record: #184=ADVANCED_FACE ('Corps principal', (#183), #158,.T.) where #183 is a pointer to the face-bounds that bound face #184 and #158 is a pointer to the description of it surface type. Face-Bounds can be of outer face bound (#183=FACE_OUTER_BOUND ('',#178,.T.)) or inner face bound.



(a) (b)

Figure 2: A sample part for explanation of STEP format

Each face is delimited by a Loop of Edges #178 =EDGE_LOOP('',(#179,#180,#181,#12))) formed by a set of Oriented_Edges (#179=ORIENTED_EDGE('',*,*,#141,.F.)).

Each of ORIENTED EDGE is defined by EDGE CURVE (#141=EDGE_CURVE('',#133,#140,#138,.T.) completely defined firstly by its Vertex points and coordinates associated to these vertices:

#141=EDGE_CURVE('',#133,#140,#138,.T.) ;
#133=VERTEX_POINT('',#132) ;
#132=CARTESIAN_POINT('Vertex',(0.,30.,15.)) ;
#140=VERTEX_POINT('',#139) ;
#139=CARTESIAN_POINT('Vertex',(55.,30.,15.)) ;
And secondly by it director vector (origin and direction):
#138=LINE('Line',#135,#137) ;
#135=CARTESIAN_POINT('Line Origine',(27.5,30.,15.)) ;
#137=VECTOR('Line Direction',#136,1.) ;
#136=DIRECTION('Vector Direction',(1.,0.,0.)) ;
The surfaces relatives to Advanced_Faces can be of planar, cylindrical, spherical or other geometrical form. The surface record of planar surface #158 is given by:
#158=PLANE ('Plane', #157);
#157=AXIS2_PLACEMENT_3D('Plane Axis2P3D',#154,#155,#156) ;
#154=CARTESIAN_POINT('Axis2P3D Location',(0.,30.,15.)) ;
#155=DIRECTION('Axis2P3D Direction',(0.,0.,-1.)) ;
#156=XDIRECTION('Axis2P3D XDirection',(0.,1.,0.)) ;

Where #157 is a pointer to the local coordinate system attached to the PLANE. The local system is given with respect to the global system attached to the part. The origin OL of this local coordinate system is completely defined by the pointer #154 corresponding to the Vertex point (0.,30.,15.) attached to OL. The direction of the normal of PLANE is given by #155 (0.,0.,-1.) and x direction necessaries to define completely the local system is given by #156 ((0.,1.,0.))

The STEP file is used as the input of our feature recognition system, and a difference between the blank and finished parts can be done. A shell itself is either ideal Manufacturing Features MF or an Interacting Manufacturing Feature $\cap$MF. Moreover Most of components of Object Oriented (OO) structure used for defining shell in their sub-entities (Faces, Edges, Vertex) in CAD STEP file seem to be very suitable for representing Manufacturing Feature. Conservation of these entities can only create certain homogeneity between CAD STEP file and CAM feature recognition system and ensure a quick intra-changeability between these two systems.

Reading of STEP file and construction of object-oriented data structure using C++ programming follow the flowchart of figure 3.

3.2 CAD INPUT FILES OF PART DESIGN

There are two manners to represent a manufacturing feature:

1) Volumetric feature that can be defined as a subset of volume swept by a cutter in a machining operation [17].

2) Surface feature defined by the set of created-boundaries surfaces by the machining operation [17].

A volumetric feature, by its massive nature, involves in its representation in addition of its proper defining surfaces, the surfaces that share with the blank. We believe that surface representation of feature is very consistent because it seems that the surface environment of a feature can always change following the volume geometry of machined part as shown by table 1 (plans, cylinder, inclined-plans and can be formed by gauche surfaces or a combination of surfaces of different geometrical types) and the main characteristics of manufacturing features that still stable in the definition of a manufacturing feature are those linked with its constituting surfaces. In another view, all manufacturing features A, B and C given in table 1 can be seen as equivalent just by only substituting the set of surfaces surrounding the feature by an equivalent spherical surface D.

3.3 TOPOLOGICAL CHARACTERIZATION OF MF

In Attributed Adjacency Graph AAG, recognition of MF is done by scanning the attributed graph and searching for sub-graph corresponding to predefined MF (step, slot, hole…). Therefore, this methodology still confined by the graph isomorphism and so cannot handle non predefined features. Moreover, the face-edge representation of MF used in AAG appears to include a non-restructured data that became very difficult to establish a unified representation of different type of MF. Also, this method fails with convex manufacturing feature that do not contain any concave edge such as inclined features.

The objective of this section is to review the examination of Adjacency Attributed Graph AAG in order to search the common characteristic point for different types of MF to unify their representation. It must be underlined that this representation will be an important step towards the extraction of MF without recourse to MF database

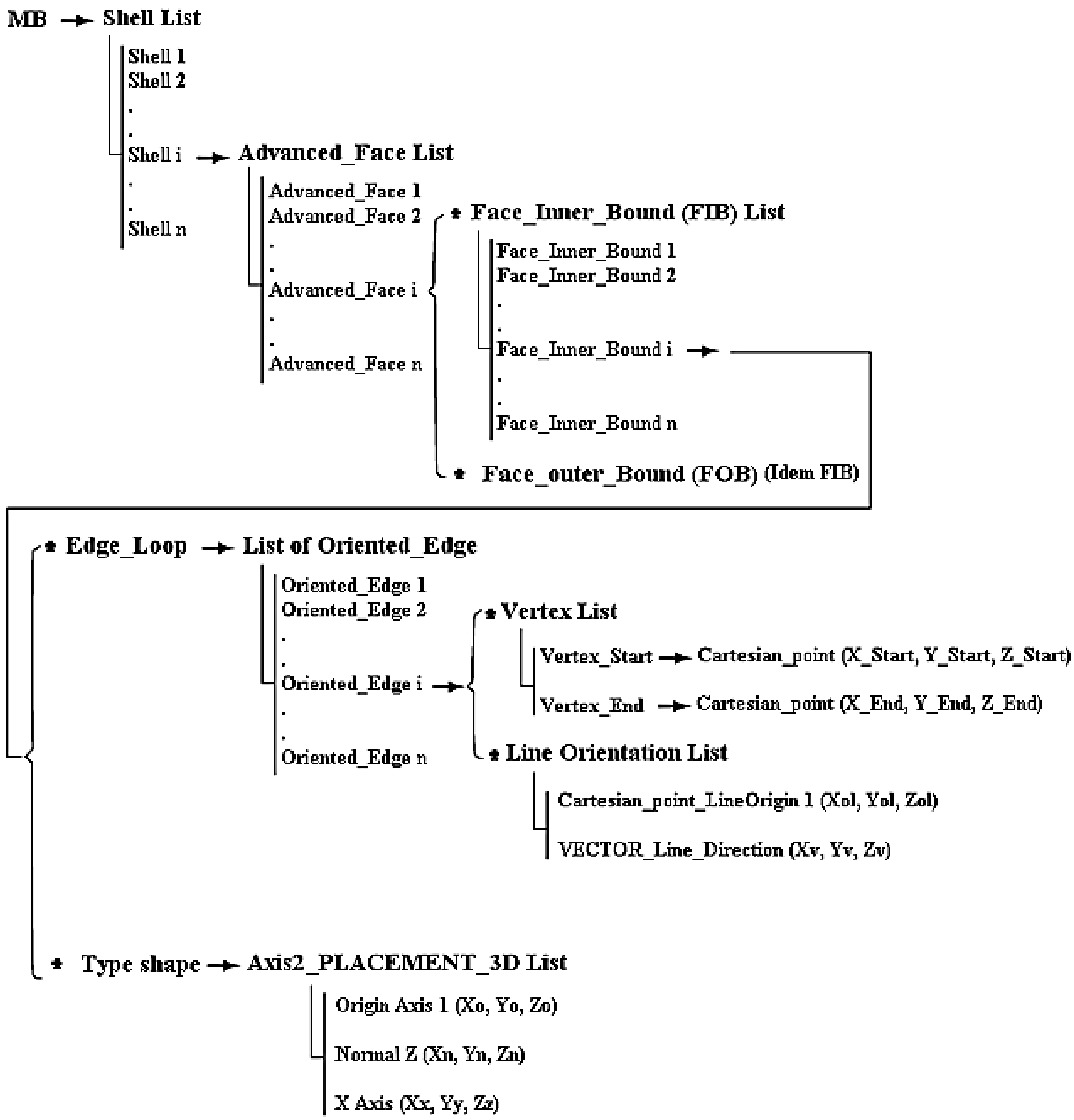


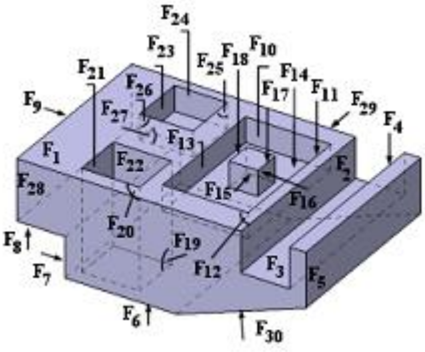
Figure 3: Translation of CAD data files into object-oriented data structure

if needed. Considering the manufacturing features given in figures 4a and 4b. The independency of definition of each MF in AAG representation made itself in the abstraction the common concept from which each manufacturing feature can be built.

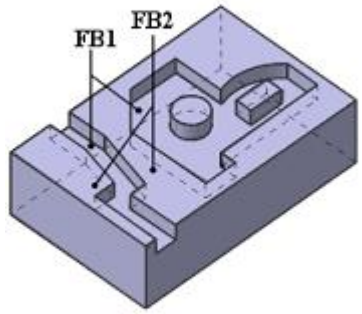
By inspecting this step and the slot of figure 4a we can conclude:

- Both step and slot begin at F28 and finish at F29. These two faces are called Faces Bases FB. A Face Base FB can be defined as the face in which rest all concave edges of MF.
- Both Step and slot are constituted by a series association of faces: F8-F9 in case of step and F2-F3-F4 in case of slot. We called this type of association Open Chain of Faces OCF.

A MF can be obtained also by a closed series association of faces as can be seen in pocket (defined by F19, F20, F21, F22). The BFs of this association are F1 and F6. This kind of association is denoted as Closed Chain of Faces CCF. In case of the web, MF is seen to be two CCFs: F11-F12-F13-F14-F11 and F15-F16-F17-F18-F15 parallel associated. We call this construction scheme of MF



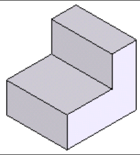
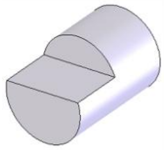
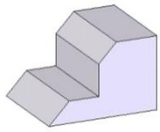
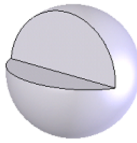
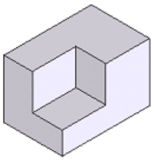
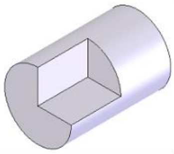
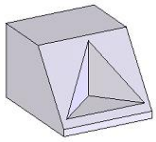
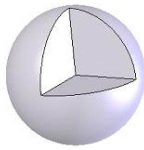
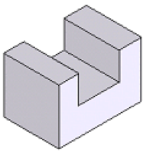
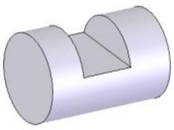
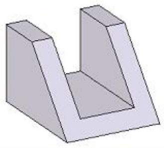
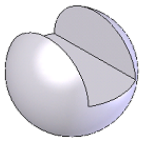
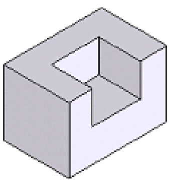
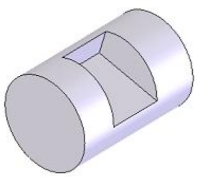
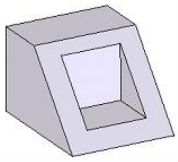
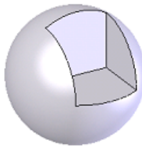
(a)



(b)

Figure 4: Simple parts for comparison of AAG and CF-BF representations

Table 1: Standard features delimited by various geometries of adjacent surface

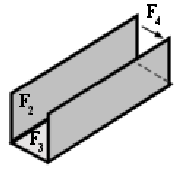
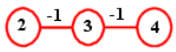
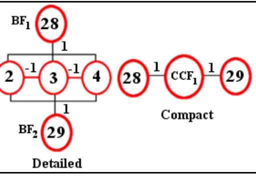
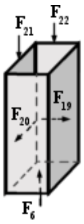
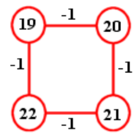
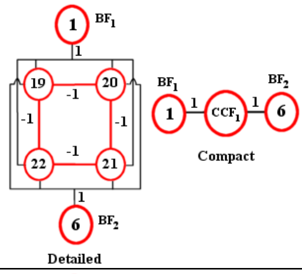
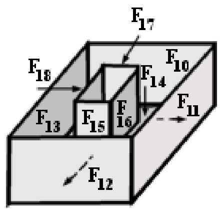
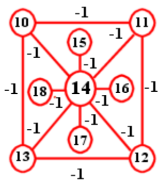
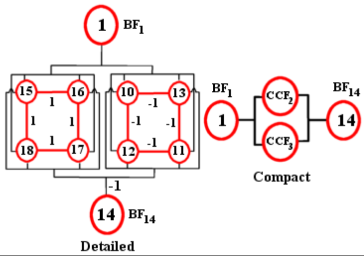
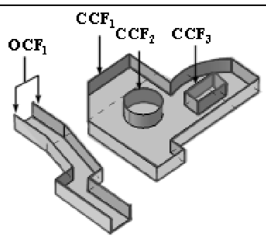
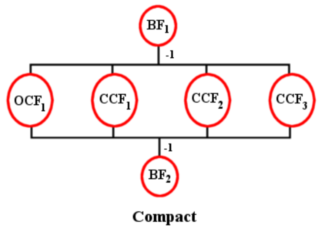
Feature	A	B	C	D
Step				
B-step				
Slot				
B-slot				

the Chains Faces–Base faces CF-BF representation. Table 2 gives a simple comparative study between the conventional AAG and current CF-BF representation. It is important to underline that graphs of standard MF such as step, slot and web were more defined and can be found in predefined libraries. In contrary, Complex Manufacturing Features are evaluative and variant following the cutting in the part. But this evaluation in our case can be controlled by the concept of structured CF-BF. For example CMF of the part of figure 4b can be easily represented and identified by it compact structured graph defined in the last line of table 2: there is four chains CCF1, CCF2, CCF3 and OCF1 parallel associated between two faces FB1 and FB2.

3.4 GEOMETRICAL CHARACTERIZATION OF A MANUFACTURING FEATURE

The definition of MF based in graph is not sufficient to identify completely the MF of the part since the graph represents really a class of manufacturing features that share the same topologic characteristic but can be geometrically distinct [18]. Geometrical conditions between faces of features such as perpendicularity and parallelism are so necessaries to arrive to a uniqueness of definition of MF. Moreover, there is a case, when certain geometrical conditions are not satisfied between MF faces, as a result, the machining of MF become impossible.

Table 2: A comparison between AAG and simple CF-BF representations

MF	SURF.REP	AAG	CF-BF
Slot			
Pocket			
Web			
CMF		-	

IV. METHODOLOGY FOR IDENTIFICATION OF MF

4.1 CONCEPT OF IMPERFECT MANUFACTURING FEATURE

Let MF1, MF2,… MFn be n interacting features of the part. Because the independency of Manufacturing features, obtaining the resulting interacting manufacturing feature ∩MF requires n machining operations.

After the nth operation, all interacting domains are devoured (missed) but each MF leaves a portion that can be considered as fingerprint of MF.

This remaining Imperfect entity is denoted Imperfect Manufacturing Feature IMF (figure 5). From a mathematical point of view, the relations that link the interacting manufacturing features IMF following n sequencings of machining operations OP can be formulated as following:

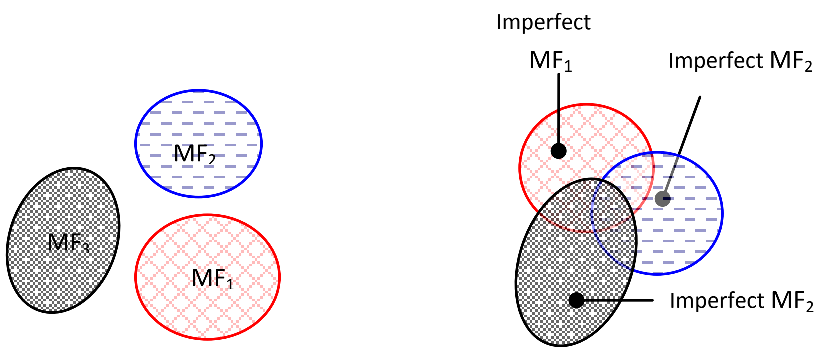


Figure 5: Scheme of Imperfect feature obtained by n interacting domains

$$\mathrm{OP}_1 : \mathrm{MF}_1 = \mathrm{IMF}_1$$

$$\mathrm{OP}_2 = IMF_2 + \overbrace{\mathrm{MF}_1 \cap \mathrm{MF}_2}^{U_2}$$

$$OP_3 = IMF_3 + \overbrace{\mathrm{MF}_3 \cap \mathrm{U}_2 + \mathrm{MF}_3 \cap (\mathrm{MF}_1 + \mathrm{MF}_2)}^{U_3}$$

$$OP_4 = IMF_4 + \overbrace{\mathrm{MF}_4 \cap \mathrm{U}_3 + \mathrm{MF}_4 \cap (\mathrm{MF}_1 + \mathrm{MF}_2 + MF_3)}^{U_2}$$

.

.

.

$$OP_i = IMF_i + \mathrm{MF}_i \cap \sum_{J=1}^{i-1} U_j + \mathrm{MF}_i \cap \sum_{j=1}^{i-1} MF_j$$

$$= IMF_i + \mathrm{MF}_i \cap MF_1 + \mathrm{MF}_i \cap \sum_{j=1}^{i-1} U_j + MF_j \qquad (1)$$

The maximal intersection number In of these n interacting features is given by:

$$\mathrm{I}_n = (\mathrm{n}-1) + \frac{(\mathrm{n}-1)(\mathrm{n}-2)}{2} + \sum_{\mathrm{i}=0}^{\mathrm{n}-3} \sum_{\mathrm{k}=\mathrm{i}+1}^{\mathrm{n}-3} \frac{(\mathrm{n}-\mathrm{k}-1)(\mathrm{n}-\mathrm{k}-2)}{2}$$

$$= -\frac{3}{4}\mathrm{n} + \frac{23}{24}\mathrm{n}^2 - \frac{1}{4}\mathrm{n}^3 + \frac{1}{24}\mathrm{n}^{24} \qquad (2)$$

It is clear from the last formulated equation (2) that if the number of imperfect interacting features IMF (the portion of feature that remains in the part) is known, the maximal interaction number of these MFs can be also known. However, it seems impossible to recoup exactly the real MFs used to build the resulting $\cap$MF because a number of faces of these effective MFs are completely missed by intersections. We attempt from the next analysis of a simple interaction $\cap$MF between two blind-pockets (figure 6) and two steps (figure 7), situated in two different local systems, to dress a clear methodology for recouping the lost portions of each IMFs.

Considering the first analysis of the part in figure 6. Supposing in machining operation OP1 we cut the blind-pocket MF1(FB, F1 F2, F5, F6, F8) and in the second operation OP2 we execute MF2(FB, F3 F6, F7, F4, Ftmiss). Before OP2, Although the Missed Intersecting Volume MIV(FB, F2, F3, F4, F5, F6, Ftmiss) defined by MF1$\cap$MF2 are completely missed, by considering, in the first step, the changes induced by the interaction to MF1, it is important to underline the following points:

- It exist always a Base Face FB from which MF can be rebuilt. In this case FB of MF1 and MF2 is the same but in general case (see figure 6), each MF possess its proper FB.
- Only the set of Frontal Face is totally missed by the interaction. This set of faces is denoted as Ftmiss.
- Lateral Faces LF (F3, F4, F6, F7) are not completely missed but just truncated and consequently the geometric types of theses surfaces are completely defined.
- It exist always a set of Edges (BI1, BI2) that mark the Begin of Interaction.
- The exact End of the Interaction (case of MF1) which is materialised by the position of Ftmiss cannot be predicted. However, it is important to precise that the position limited of Ftmiss can be always

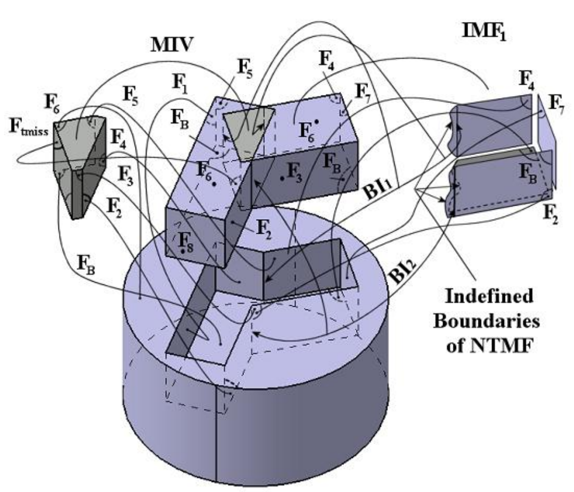


Figure 6: Analysis of a component obtained from the interaction between two Blind-pockets

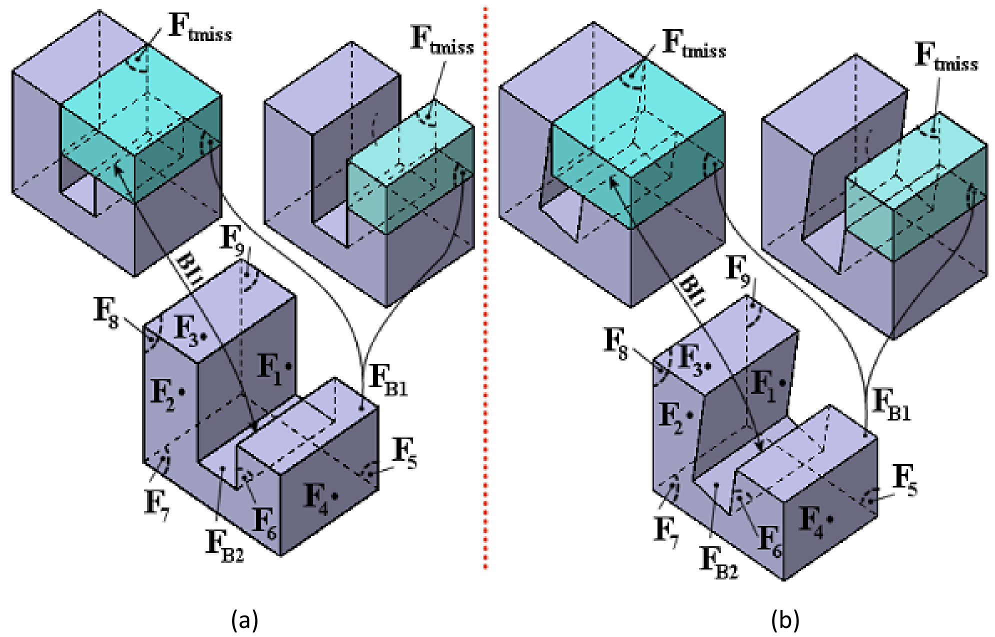


Figure 7: limit location of the Face Missed Ftmiss by the interaction between two MF: slot and step

determined and a suitable virtual Face FV that substitutes Ftmiss can be therefore stored.

For this purpose, considering two IMFs between step and slot taken in two distinct configurations (Figure 7). In case (a) of figure 8, the local systems related to the step and B-slot are parallel but the position of the FB1 of step is different from FB2 of B-slot. In figure 7, the orientation of the two local systems is also different.
Although, from figure 7a, F1 can be seen as the limit of Ftmiss, In general case, as shown by the figure 7b, Ftmiss is not all the time merged with F1 But its limit is given by subtracting the extended face of FB situated between the begin of interaction BI1 and face obtained by projecting the Frontiers Face FRF F1 on FB plane's called PF1/FB. The domain obtained by this subtraction, that materializes the possible position of that Virtual Face called FV is denoted as Domain of Construction DC. Note: F1 defines the set of adjacent surfaces of FB1 that are not beyond imperfect manufacturing feature IMF1.
Finally the model for rebuilding IMF can be described by figure 8: there is a microscopic volume dv that put down on a Face Base FB of IMF surrounded by a set of Frontiers Faces FRFs. This volume is susceptible to grow within a Domain Construction. In addition of its Faces, IMF can use the Frontiers Faces as boundaries of the growth volume if it satisfies the required conditions to build a suitable MF, else,

a set of virtual Faces is constructed accordantly with these conditions.

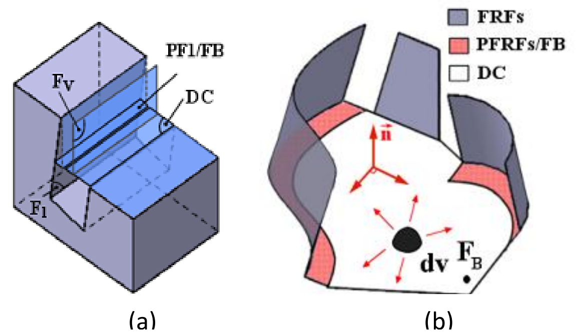


Figure 8: Model for rebuilding interacting features: a) virtual Face that materialize the possible position of Ftmiss, b) Domain of construction by taking Account the faces boundaries surrounding the IMF

4.2 PERFECT MANUFACTURING FEATURE MF, COMPLEX MANUFACTURING FEATURE CMF AND IMPERFECT MANUFACTURING FEATURE IMF

Before describing the methodology followed for recuperation the parties lost by interactions, it will be important at this stage to define the following terms: Perfect Manufacturing Feature MF, Complex Manufacturing Feature CMF and Imperfect Manufacturing Feature IMF.
Perfect MF designates here is not the classical elementary predefined MF such as step, slot and web but it is defined as a simple OCF or CCF that rest on two Bases Faces and satisfy the recurred geometrical condition to be machined. To be perfect, also the surrounded faces of OCF or CCF must be blank faces of the part.
A Complex Manufacturing Feature is an association

parallel/series of OCF and CCF that rest on a Base Face FB. It is obvious that a MF is a subclass of CMF but it important to precise that, in spite that many authors consider a CMF as ∩MF in fact that this last can be subdivided in several MF that can be seen as interacting MF, it well be noticed that all MFs that constitutes the CMF share the same BF and consequently can be machined in a single operation. So the first main characteristic that differentiates MF and CMF from ∩MF is that in the case of ∩MF, each MF that participate in the interaction, possesses it proper Bases Faces. The second is in the CMF (same of MF), except BF, all faces surrounding MF or CMF are blank faces.
So:

A MF is considered as PMF if:

I. It Exist an OCF or CCF of faces that rest on BF.
II. All surfaces that constitute the feature are material surfaces.
III. All edges shared by it adjacent surfaces are material edges.
IV. The topological and geometrical conditions between the surfaces that define the feature are valid.
V. All faces except those for defining MF are blank (stock) faces.

A feature is called CMF if:

I. It Exist a group of OCF or/and CCF of faces that can be parallel/series associated between two BFs.
II. Conditions i to v of perfect MF are also satisfied for each OCF or CCF.

A feature is called IMF if:

It exist a face base FB that is linked with a set of faces Fi (at least FB itself):

a. It exist an OCF or a CCF that rest on BF for which the topologic and geometric criterions are satisfied but it exist at least one face that delimits OCF or CCF which is not a blank face.

Or

b. If it exist a set of material surfaces Fi linked with FB that satisfies the following conditions
I. Geometrical relations with respect to BF are satisfied.
II. It is not possible from the existent material Fi to achieve the rebuilt of any OCF or CCF without recurring to virtual Faces Fv that try to substitute the Faces missed Ftmiss by the interaction.

4.3 ADJACENCY RELATION BETWEEN FEATURE FACES.

4.3.1. CLASSIFICATION OF EDGES

The concavity test is based on the angle between two adjacent faces [15, 20]. Depending on the angle between two adjacent faces, an edge can be classified as convex, concave, smooth-convex or smooth-concave (figure 9).
The concavity test for a given common edge E between two faces F1 and F2 (figure 9) is performed based in information from STEP file of the part and is performed as follows:

1) Identification of the normal direction $\vec{n}_1$ and $\vec{n}_2$ of two planes that support F1 and F2.
2) Determination of orientation of Face-Loop using the right hand rule
   - The orientation of Outer-Loop with respect of the normal is in counter-clockwise direction.
   - The orientation of Inner-Loop with respect of the normal of face is in clockwise direction.
   - Determination of orientation $\vec{u}_E$ of the common shared edge E between the adjacent faces F1 and F2 with respect of orientation Edge-Loop of F1.
   - Calculate the cross product $\vec{c} = \vec{n}_1 \wedge \vec{n}_2$
   - Calculate the projection vector of $\vec{c}$ on the directional vector of the common edge $\vec{u}_E$: $d = \left(\vec{n}_1 \wedge \vec{n}_2\right).\vec{u}_E$
3) Determination of concavity type:
   - If d >0 the shared-edge E between F1 and F2 is concave.
   - If d <0 the shared-edge E between F1 and F2 is convex.
   - If d=0 the edge is smooth.

4.3.2. GEOMETRICAL POSITION

The orientation of F1 with respect of it adjacent face F2 is given by the scalar product between the normal vectors of this two faces: $O_{F1/F2} = \vec{n}_1 . \vec{n}_2$

- If OF1F2 =0 F1 and F2 coplanar-faces
- If OF1F2 =1 F1 and F2 are perpendiculars
- If OF1F2>0 F1 and F2 obtuse-angled
- If OF1F2<0 F1 and F2 acute-angled

4.4 METHODOLOGY OF MF RECOGNITION

The feature recognition system can be subdivided in two distinct algorithms: The first

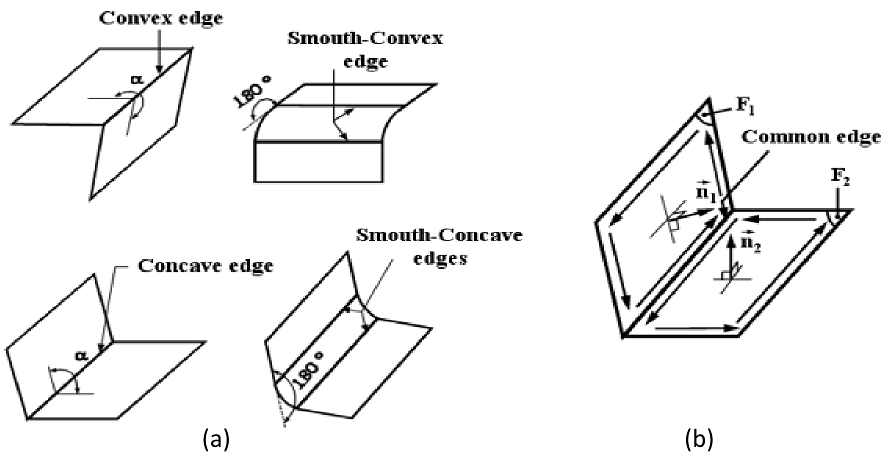


Figure 9: Edge classification (a) and concavity test of shared edge between two adjacent faces F1 and F2 based on right hand rule (b).

algorithm (figure 10) permits extraction of perfects MF and CMF and the second (figure 11) Recoups ∩MF. The mains steps of this system can be explained from the simple part associated with these two algorithms.

Algorithm 1:
Step 1: Determination of difference between the finished part and the blank: Machined Block MB
Step 2: Generation of STEP file of MB.
Step 3: Reading of STEP file using C++ programming and generation of Object-Oriented Data structure following the chart represented in figure 3: List of shells, List of Faces constituting each shell…
Step 4: Adjacency relations between all faces
For each shell of part do
Determination of class of adjacency of each face of the shell
Concavity of shared edges between adjacent faces
Relative geometrical orientation between this two faces
Identification of all Base Faces BF
For each BF
Research of CCF and OCF that rest on BF
If all surrounded-faces of OCF or CCF are a blank faces then
If number of OCF or CCF is equal to one then
There is a perfect MF found else
Else there is a perfect CMF found
Else Generation of Imperfect Features IMF and go to Algorithm 2
End do

Algorithm 2:
Step 8: Identification of all Frontier Faces
Step 9: Elimination of all faces situated bellow BF.
Step 10: Projection of all Frontier Faces on BF and determination of Domain of Construction DC.
Step 11: Classification of Frontier faces neighbour to neighbour.
Step 12: Instauration of virtual Faces to complete OCF or CCF.
Step 13: Extraction of MF that constitutes ∩MF.

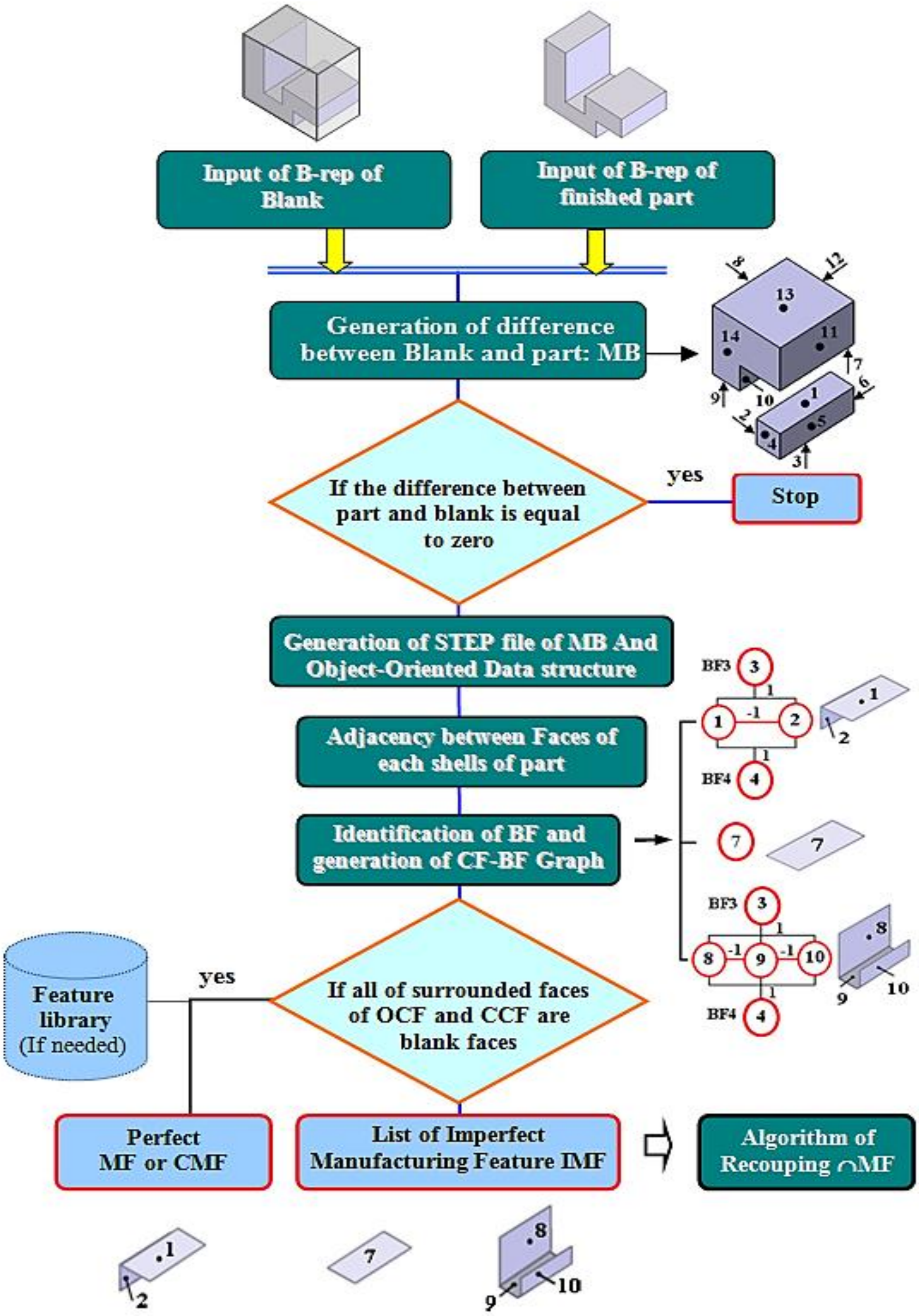


Figure 10: Algorithm 1: Flowchart for extraction of perfect MF and CMF

V. APPLICATION

The sample application is limited to prismatic parts but the algorithm can be extended and applied to complex parts. The example part shown in Figure 12 is the same of that presented in figure 9 by Gao and al. [16] but the orientation of local system axis of the Interacting Manufacturing Features is taken to be different. This part is used only to clarify the method developed and not to test its limitations. The algorithm 2 permits successively to recoup the missed Manufacturing Features that participate in the interaction and generates five combinations of possible machining of the part (figure 13). All combinations of MF for each output are constituted by three Manufacturing Features. It should be noticed that the five combinations are really independents and there is no redundancy found.

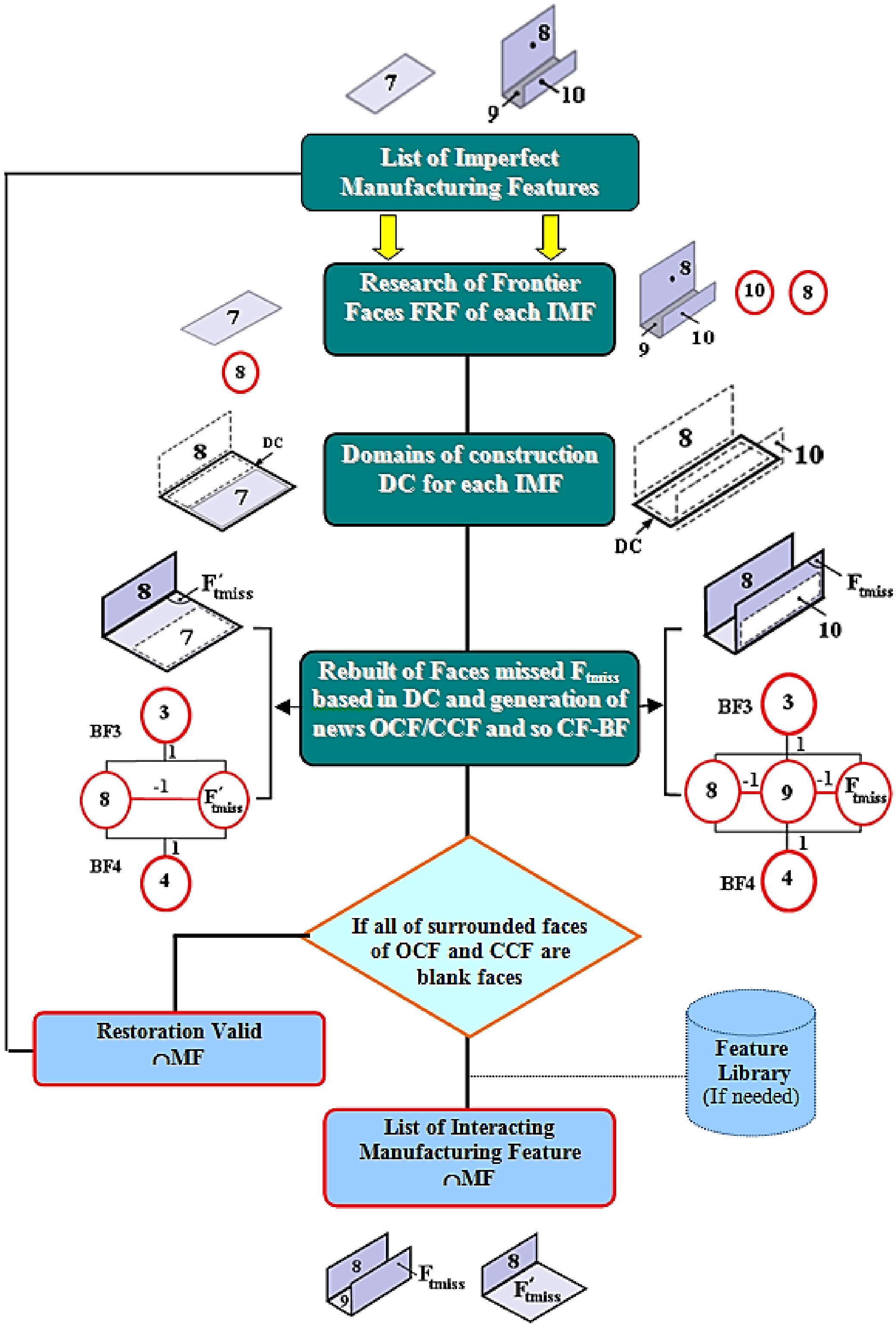


Figure 11: Algorithm 2: Recouping of the Interacting Manufacturing Features ∩MF

Multiple interpretations of MF are considered among the desired fertilities of a given recognition system, because it offers, upon request, the possibility to select a specific sequencing among the generated varieties of the system [16].

However, generation of alternative interpretations can lead to a problem of combinatorial explosion when interactions between features become more complex [17]. In our case this problem is greatly avoided firstly by the notion of IMF introduced and secondly by choosing the largest Domain of

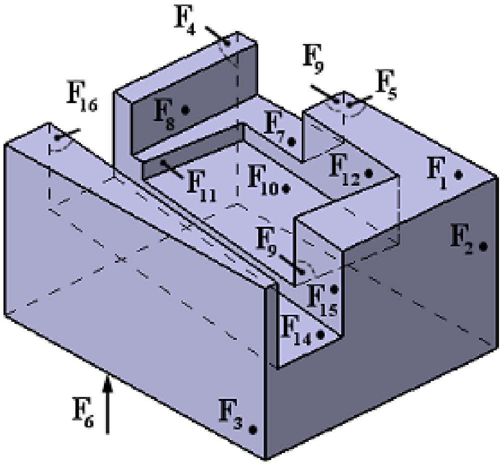


Figure 12: Sample workpiece for application of Recouping $\cap$MF using Algorithm 2

Construction DC by a judicious analysis of the Frontier Faces of FB. But it must be admitted that it is very difficult to conclude on the optimized sequencing to be adopted and so an algorithm to optimize this processing system seems to be necessaries. This issue will be projected among the prospects for this work.

VI. CONCLUSION

In spite of an immense research efforts done in automatic feature recognition AFR, which are behind the development of many industrial systems in CAPP activities, it remain a fertile of researcher due to the complexity of components that participate in its constitution, such as diversity of CAD models, Complexity and variety of geometrical forms of parts, characteristics of machining Processes and tools, technological nature of the domain (expertise), mathematical algorithm frame used, the complexities and some difficulties that result from the extraction of manufacturing feature itself, principally when this feature have a complex geometry, complex environment or when the feature is missed or transformed by an interaction with its adjacent features.

In this paper we have qualitatively described a new simple system for automatic recognition of manufacturing feature ARMF, that can handle both isolated and interacting features from the STEP file of a machined part. A feature in this model is seen as a Base Face FB in which rest on a set of Opened Chains of Faces OCF or Closed Chains of Faces CCF. To be perfect, all boundaries surfaces of OCF and CCF must belong to blank faces of the part. In contrary the MF is considered as an Imperfect Manufacturing Feature IMF. In order to recoup all Missed Faces of IMF, an elementary volume dv is planted on Face Base FB of IMF. This elementary volume is susceptible to grow following all directions within the Frontier Faces of FB to be transformed to a perfect MF, but this growth is conditioned by the possible machining of the final generated volume. So, an operator, associated with dv, inspects all Frontier Faces by analysing its relative localisation with respect of FB. A valid construction is performed by projecting all Frontiers Faces on FB and then a Domain of Construction of Faces is performed. Frontier Faces that satisfie the recurred geometrical conditions are conserved and new virtual Faces are created until all surrounded faces of CCF and OCF are blank faces. Despite the fact that this method has the ability to give multiple interpretations of features, an algorithm for optimisation of the process still necessaries and will be planed among the optics envisaged for the next works.

REFERENCES

[1] Chee Fai Tan, V.K. Kher and N. Ismail, Design of a Feature Recognition System for CAD/CAM, Integration, *World Applied Sciences Journal* 21 (8) (2013).

[2] J. C. E. Ferreira, D. Vivian, Feature recognition in axisymmetrical parts modeled by solids in an Internet-oriented CAD/CAM system, *Journal of Materials Processing Technology* 179 (2006) 260-267.

[3] Hesam Samarghandy, Ye Li, Detecting re-design area for increasing manufacturability of drilling and three-axis pocketing operations, *The International Journal of Advanced Manufacturing Technology* Vol. 69 Issue 1-4 (2013) 337-349.

[4] S. Srinivasa Rao , B. Satyanarayana, M.M.M. Sarcar, Automated generation of NC part

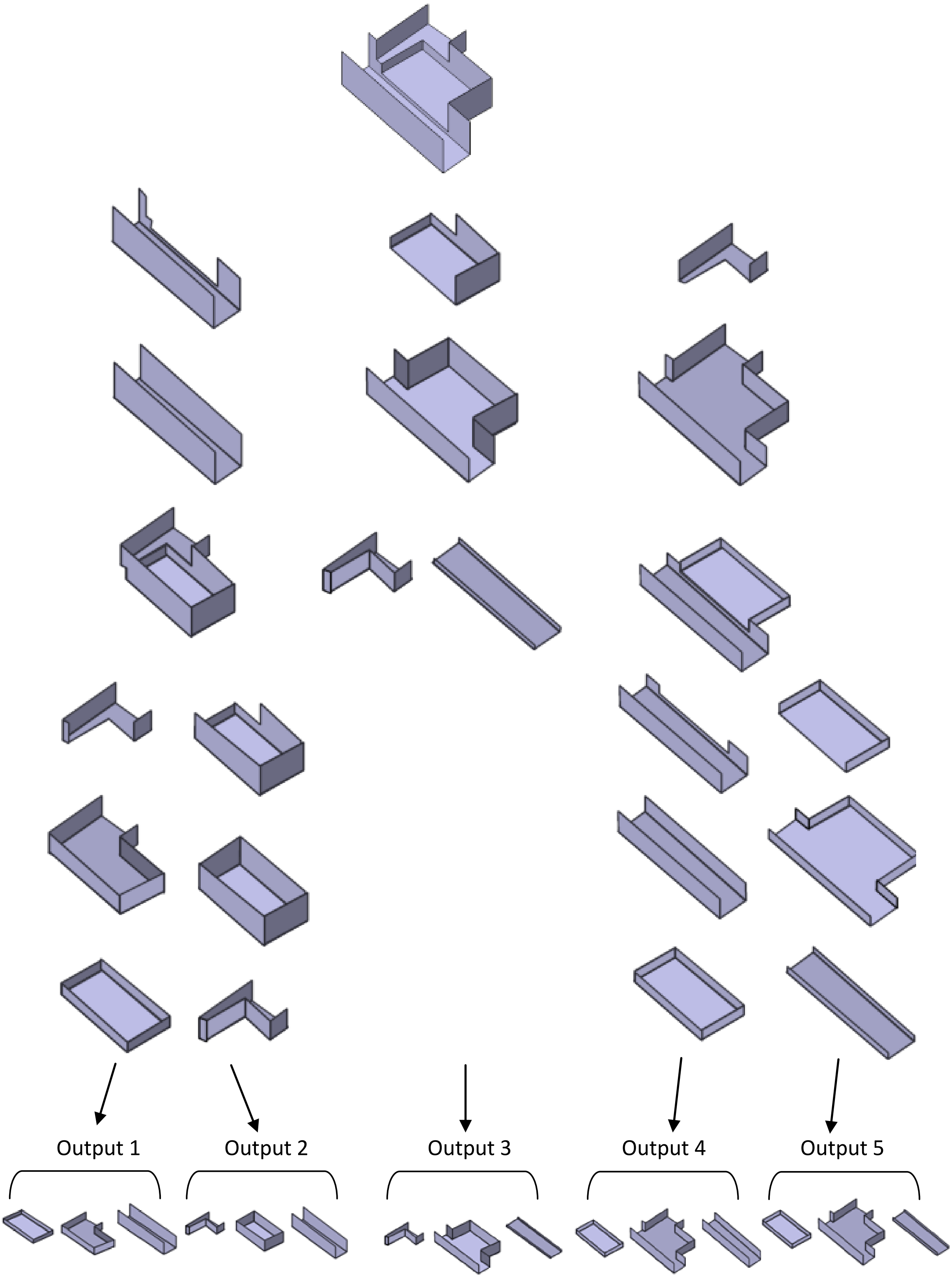


Figure 13: Manufacturing Feature derived from application of Algorithm

2 programs for turned parts based on 2-D drawing image files, *International Journal of Production Research*, Volume 50, Issue 12, (2012) 3470-3485.

[5] Joshi S. et Chang T.C., "Graph based heuristics for recognition of machined features from a 3-D solid model", *Computer Aided Design*, vol. 20, pp. 58- 66, 1988.

[6] Aun Yong Bok, Mohd Salman Abu Mansor, Generative regular-freeform surface recognition for generating material removal volume from stock model, *Computers & Industrial Engineering* 64 (2013) 162–178.

[7] Cheol-Soo Lee, Jae-Hyun Lee, Dong-Soo Kim, Eun-Young Heo, Dong-Won Kim, A hole-machining process planning system for marine engines, *Journal of Manufacturing Systems* 32 (2013) 114– 123

[8] V. Sundararajan, P.K. Wright, Volumetric feature recognition for machining components with freeform surfaces, *Computer-Aided Design* 36 (2004) 11–25.

[9] Sean Tessier, Yan Wang, Ontology-based feature mapping and verification between CAD systems, *Advanced Engineering Informatics* 27 (2013) 76–92

[10] N. Ismail, N. Abu Bakar, A.H. Juri, Feature recognition patterns for form features using boundary representation models, *International Journal of Advanced Manufacturing Technology* 20 (2002) 553–556.

[11] Songqiao Tao, Zhengdong Huangb, Lujie Ma, Shunsheng Guo, Shuting Wang Youbai Xie, Partial retrieval of CAD models based on local surface region decomposition, *Computer-Aided Design* 45 (2013) 1239–1252.

[12] Xionghui Zhou, Yanjie Qiu, Guangru Hua, Huifeng Wang, Xueyu Ruan, A feasible approach to the integration of CAD and CAPP, *Computer-Aided Design* 39 (2007) 324–338.

[13] L. Kyprianou, Shape classification in CAD, PhD Thesis, Univ. of Cambridge (1980).

[14] X. Xu, S. Hinduja The recognition of rough machining features in 21/2D components, *Computer-Aided Design* 30 (7) (1998) 503-516.

[15] Mamadou Sya, Christian Masclea, Product design analysis based on life cycle features, *Journal of Engineering Design*, Vol. 22, Issue 6 (2011) 387-406.

[16] S. GAO, J. J. SHAH, Automatic recognition of interacting features based on MSCG, *Computer-Aided Design* 30 (9) (1998) 727-739.

[17] A.H. Zulkifli, S. Meeran, Decomposition of interacting features using a Kohonen self-organizing feature map neural network, *Engineering Applications of Artificial Intelligence* 12 (1999) 59-78.

[18] B. Babic, N. Nesic, Z. Miljkovic, A review of automated feature recognition with rule-based pattern recognition, *Computers in Industry* 59 (2008) 321–337.

[19] A. AzwanIskandar, M. Jamaludin, M. TaibZulkepli 2002 *Jurnal Mekanikal* Disember 2003, BU. 16, 31 -46.

[20] N. Ismail, N. Abu Bakarb, A.H. Juri, Recognition of cylindrical and conical features using edge boundary classification, *International Journal of Machine Tools & Manufacture* 45 (2005) 649–655.