Journal of Theoretical and Applied Information Technology
15th May 2023. Vol.101. No 9

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

A FEATURE RECOGNITION SYSTEM FROM STEP AP203 NEUTRAL FILE, BASED ON STEP AP224 MANUFACTURING FEATURES

OUSSAMA JAIDER[1], NIHAD AGHBALOU [2]
[1] Faculty of Sciences and Technologies of Tangier, Morocco
[2]EIGSI Casablanca

E-mail: [1] jaider.oussama@outlook.com, [2] nihad.aghbalou@eigsica.ma

ABSTRACT

Geometric modeling has been used for many years as a design tool for CAD (Computer Aided Design) systems. However, data definitions provided by these systems cannot be used directly by a computer-aided process planning system. As a result, the concept of Automatic Feature Recognition (AFR) has been introduced to integrate CAD and downstream applications such as CAPP (Computer Aided Process Planning). However, even though a feature recognition system is capable of recognizing and extracting manufacturing features from a neutral file such as STEP AP203, these features are still defined by geometric and topological low-level data and thus, lack of standardization. For this reason, STEP AP224 application protocol has been developed to provide a standard for both implicit and explicit representations for machining features. Nevertheless, recognizing machining features of AP224 from a neutral file such as STEP AP203 is still the subject of several research. The main purpose of this paper is to explain the methodology of converting manufacturing features obtained by a recognition system we have developed previously, to machining features defined according to ISO 14649. The EXPRESS modelling, Python language and the STEPCode library are the main tools used to map parameters of objects representing manufacturing features obtained by the AFR system, to geometric and dimensional attributes of classes representing features of ISO 14649. An example part was tested to show the results of the developed application, called CAPP-Turn.

Keywords: *AFR, STEP AP224, ISO 14649, CAD/CAPP/CAM*

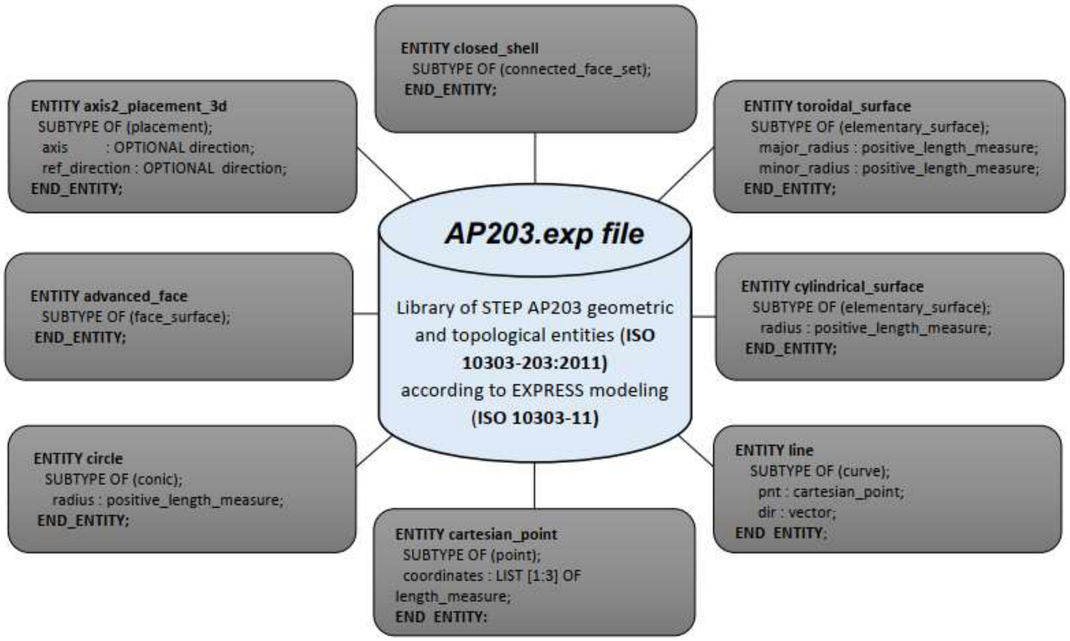
1. INTRODUCTION

In traditional industrial practice, product knowledge is stored in drawings, manuals and supplier data sheets. Today, such methods are largely replaced by CAD models and various databases. CAD systems use 2D or 3D geometric modeling approaches to represent product design. As a result, design information provided by these systems is implicit and represented in terms of low-level primitives, thus limiting access to a complete manufacturing analysis [1, 2]. One of the major challenges of computer-integrated manufacturing (CIM) is to make common languages between CAD and CAM, through CAD/CAM integration. Therefore, design information provided by CAD systems needs to be translated into explicit manufacturing information such as machining features, which can be directly used by various downstream applications [3]. Then, these features serve as a link between CAD and downstream applications (CAM, process planning, inspection, assembly ...). Machining features maintain a high level of abstraction from the description of the part. This means that machining features provide not only geometric and topological entities, but also dimensions, tolerances, materials, surface roughness, etc. In this sense, feature creation techniques have been developed to represent this high-level design information, and significant research efforts have been made on this subject. The most used feature creation techniques are grouped into two approaches, namely feature-based design and automatic feature recognition [4]. Each of these techniques has its own advantages and drawbacks, but the difficulty faced by both approaches is the lack of implicit and explicit representation standards for an unmanageable number of predefined features [5].

To overcome these problems and expand the scope of CAD systems beyond geometric modeling, and a little further than feature modeling, the introduction of the ISO application protocol AP224 [6] has been specified for the definition of mechanical products in terms of machining features. STEP AP224 provides a universal feature library with technological attributes that are familiar to designers. However, recognizing AP224 features from an AP203 data file is still being a major research topic [7, 8]. This paper aims on one hand to present a technically detailed improvement of the work done by Jaider et al [3], regarding automatic feature recognition from STEP AP203 file. This enhancement includes the use of the EXPRESS schema as a modeling language, and Python as an object-oriented programming language to extract and handle geometric and topological data of STEP AP203 part file. On the other hand, it is to exploit these low-level geometric and topological data characterizing the recognized manufacturing features [3], in order to define in a first work, just geometric and dimensional parameters of features defined according to the STEP-NC standard. ISO 14649 has been adopted as a standard for automatic turning process planning activities, thus, turning features parameters will be defined in accordance with turning feature parameters defined in part 12 of ISO 14649, since manufacturing features definitions in this standard are fully harmonized with those of STEP AP224.This guide provides details to assist authors in preparing a paper for publication in JATIT so that there is a consistency among papers. These instructions give guidance on layout, style, illustrations, and references and serve as a model for authors to emulate. Please follow these specifications closely as papers which do not meet the standards laid down, will not be published.

2. RESEARCH WORKS BASES ON STEP AP224 MACHINING FEATURES

KANG et Al [7] propose an approach to interlink design and process planning by associating part geometry with manufacturing information in an integrated product model based on STEP neutral format. First, the EXPRESS information model for AP203 is compiled to produce C++ classes using the ROSE library. Geometric entities of STEP AP203 are extracted, interpreted, and converted into BRep data structure. The BRep entities are then translated into a Parasolid model by using Parasolid Application Protocol Interface (API) functions. After that, relevant technological information such as surface roughness, dimensional and geometric tolerances are assigned. The outcome is a geometry model with tolerance assignments, from which machining features are recognized. Together with the geometric model given are the machining features and tolerance data stored in a STEP AP224 file. Once a physical STEP AP224 file is generated, any downstream activities including process planning or inspection planning can be automated. Amaitik and Al [2] propose the idea of embedding a STEP AP224 feature-based modeler as a design tool in the existing CAD systems so as that designers can consider design and manufacturing aspects in earlier stages of production process. Their approach consists in generating a STEP AP224 data file by creating machining features instances according to STEP AP224 features parameters without any need to feature recognition algorithms. Features instances are obtained after translating the EXPRESS format for each feature according to STEP part 21 text encoding format. A 2D feature region is created by the designer based on the required geometric and dimensional parameters of a STEP AP224 feature, then a 3D feature volume is generated by revolution or extrusion of the planar profile about or along a specified axis. Once the feature is created, it is attached to the main part. Feature attachment process consists of three steps. First, the feature position is fixed by placing the feature in the specified insertion point in the Part Coordinate System (PCS). The PCS of cylinder base shape is positioned in the center of the one base of cylinder. Then, the feature orientation is located by using direction and location elements with respect to the three axes of the PCS. Finally, feature Boolean operation is performed by subtracting, intersecting, or adding the feature to the part. The major drawback of this method occurs when interacting features exist; on one hand, it is too difficult for the planner/designer to handle all features combinations, and then to choose the optimal set of features, on the other hand, defining geometric and dimensional parameters of a set of features seem to be not easy since these parameters need to be modified due to interactions, as we will see next, design features of the final part are not always the same of machining features. Chen and al [8] developed an automatic feature recognition system based on AP224 machining features, taking A STEP AP203 file as input to the system. Before any manufacturing feature is recognized, Microsoft Access has been used to create a database for processing the geometric and topological date contained in a neutral AP203 data file. A convexity test is implemented to determine the types of every edge in these data, whether it is convex or concave. This pattern of convexity of the edges is used for mapping manufacturing feature with graph-based approach and helping in manufacturing feature



```
ENTITY closed_shell
  SUBTYPE OF (connected_face_set);
 END_ENTITY;
```

```
ENTITY axis2_placement_3d
  SUBTYPE OF (placement);
   axis         : OPTIONAL direction;
   ref_direction : OPTIONAL  direction;
END_ENTITY;
```

```
ENTITY toroidal_surface
  SUBTYPE OF (elementary_surface);
    major_radius : positive_length_measure;
    minor_radius : positive_length_measure;
END_ENTITY;
```

Library of STEP AP203 geometric and topological entities (**ISO 10303-203:2011)** according to EXPRESS modeling **(ISO 10303-11)**

```
ENTITY advanced_face
  SUBTYPE OF (face_surface);
 END_ENTITY;
```

```
ENTITY cylindrical_surface
  SUBTYPE OF (elementary_surface);
    radius : positive_length_measure;
 END_ENTITY;
```

```
ENTITY circle
  SUBTYPE OF (conic);
    radius : positive_length_measure;
 END_ENTITY;
```

```
ENTITY line
  SUBTYPE OF (curve);
    pnt : cartesian_point;
    dir : vector;
END  ENTITY;
```

```
ENTITY cartesian_point
 SUBTYPE OF (point);
 coordinates : LIST [1:3] OF
length_measure;
END  ENTITY;
```

Figure 1: Some Geometric And Topological Entities Of STEP AP203 According To EXPRESS Modeling

recognition using rule-based approach, taking into account raw stock determination and subtraction of part from stock. Lastly, the recognized manufacturing features are generated in ISO 10303 AP224 text file, which is useful for subsequent manufacturing processes.

3. GEOMETRIC AND TOPOLOGICAL DATA EXTRACTION FROM STEP AP203 FILE

3.1 Geometrical and topological entity modeling of STEP AP203 application protocol according to EXPRESS language

One of STEP's key objectives is to provide unambiguous, computer-readable representation of product data. This hypothesis is supported using the EXPRESS language. EXPRESS is a data specification language used to represent the structure of data in a specific application context. Various data models have been developed to describe product shape requirements (geometry and topology), technical drawings, and industry specific requirements. Data models contained in integrated resources and STEP application protocols are defined using EXPRESS. The STEP AP203 Application Protocol, for Configuration controlled 3D design of mechanical parts and assemblies, has been developed to ensure sharing and exchange of CAD models across the entire CAD / CAPP / CAM chain. Geometric and topological data described in the AP203 are all modeled by the EXPRESS language. Figure 1 shows the EXPRESS definition of some geometric and topological entities of STEP AP203. The EXPRESS schema describing STEP AP203 application protocol (with the extension .exp), as well as other application protocols are available on the STEPcode website [9, 10]. The STEPcode project is a collection of open source libraries, tools and resources for ISO 10303. Formerly known as STEP Class Library (SCL), the project has evolved from its origins at NIST (National Institute of Standards and Technology) into a diverse open source community helping to the improvement, accessibility, adoption and promotion of STEP for CAx tool developers, such as CAD, CAM, CAE and other tools. This project provides a robust implementation of an EXPRESS parser schema for ISO 10303 P21 files, with the ability to translate EXPRESS schemas to C, C ++ and Python classes.

3.2 Structure of STEP AP203

To have a good understanding of the structure of STEP AP203 file as well as most of its geometric and topological entities, refer to the article [3], where instances of these entities (and their attributes) in the STEP AP203 file are well explained. This is very useful for users interested in understanding the neutral file format of STEP AP203, for developers of CAPP systems, and in general for developers of applications based on this standard.

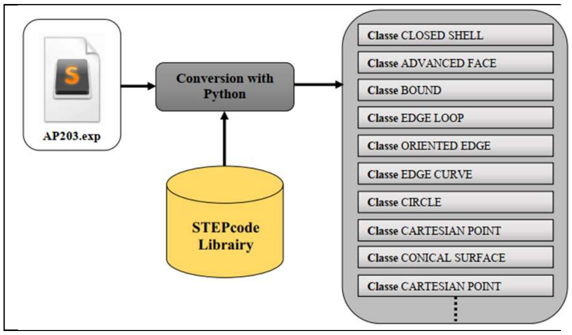


Figure 2: Geometric And Topological Entities Translation Of AP203 EXPRESS Schema Into Python Classes

3.3 Conversion of geometric and topological entities of AP203 EXPRESS schema into Python classes

The EXPRESS definitions of geometric and topological entities included in the AP203 application protocol schema are necessary for an efficient extraction of these data from a STEP AP203 file. To exploit these definitions effectively, STEPcode has been used in this work to translate geometric and topological entities modeled according to EXPRESS into Python classes (Figure 2). STEPcode includes class libraries, some of the most widely used EXPRESS schemas, some tools that work with EXPRESS, and other support libraries for these tools.

In object-oriented programming, a class declares properties that are common to a set of objects. These properties named attributes represent the state of the objects. Once features instances parameters are extracted from a STEP AP203 file, the obtained classes will be used to create objects that represent geometric and topological data of any part. An example of translation of the entity CARTESIAN_POINT (according to EXPRESS) into Python class is shown in Figure 3.

3.4 Instance parameters mapping of AP203 file to entity classes Attributes of AP203 EXPRESS Schema.

As mentioned in paragraph 3, the STEPcode library allowed us to translate geometric and topological entities contained in the EXPRESS schema of AP203 into Python classes. The next step is to define scripts for mapping AP203.stp instance parameters to attributes of the previously obtained entity classes. In computing, a mapping is an association of data belonging to one set (logical data model, production database…) with data belonging to another set (physical data model, data warehouse…) , then the data of the first set can be substituted for those of the second set, or that we can pass harmoniously from first to second. To do this, we developed with Python many mapping functions for each entity of the EXRPESS schema. And in order to correctly instantiate class objects of each entity and relations between them, the mapping functions have been designed in a nested way: a function can call several other functions. In this way, the process of extracting data from a STEP AP203 file begins by instantiating the objects of the lowest level classes until reaching the highest level entity, namely CLOSED_SHELL. An example of mapping instance parameters of the CARTESIAN_POINT entity to the object attributes of this entity is shown in Figure 4.

```
class cartesian_point(point):
    '''Entity cartesian_point definition.

    :param coordinates
    :type coordinates:LIST(1,3,'REAL', scope = schema_scope)
    '''
    def __init__( self , inherited0__name , coordinates, ):
        point.__init__(self , inherited0__name , )
        self._coordinates = coordinates

    @property
    def coordinates(self):
        return self._coordinates
    @coordinates.setter
    def coordinates(self, value):
        # Mandatory argument
        assert value != None, 'Argument "value" is mantatory and cannot be set to None'
        if not check_type(value,LIST(1,3,'REAL', scope = schema_scope)):
            self._coordinates = LIST(value)
        else:
            self._coordinates = value
```

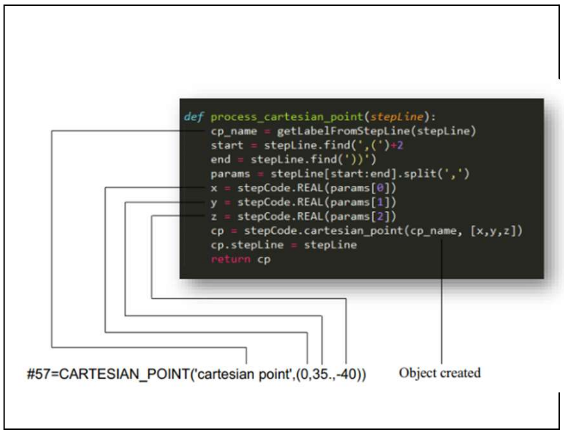
Figure 3: Translation Of CARTESEIAN_POINT Entity Into Python Class

4. AUTOMATIC FEATURE RECOGNITION (AFR) USING RULE-BASED AND ENTITY GROWING APPROACHES

The proposed system for automatic feature recognition consists in analyzing geometric and topological objects obtained by the mapping method presented in the previous paragraph. Analysis of geometric objects deals with the evaluation of centers coordinates of circles, cylinders, cones, tori, hemispheres, radiuses of circles, cylinder, etc.

Analysis of topological objects concerns the evaluation of edge loops (EDGE_LOOPs), types of surfaces, construction of edge curves, internal edge loops (FACE_BOUNDs), external edge loops (FACE_OUTER_BOUNDs), vertices, etc. The feature recognition system has a limited feature class library that we have defined but can be extended by adding other feature classes. By analyzing geometric and topological objects obtained before, the rule-based system identifies a manufacturing feature according to a number of pre-established rules (defined in the feature class), which are characteristic to the feature. The recognition method begins by identifying features composed of a maximum number of surfaces, such as grooves and recesses that we have called; Tertiary features (Figure 5). Then, features to be identified are named secondary features, and finally, primary features such as chamfers, rounds, faces and cylinders. This order was taken wisely because the system can identify for example a shoulder, and a face, or a cylinder and two faces, while it is a rectangular groove (composed by a cylinder and two planar surfaces). This seemed to us more judicious than introducing even more rules, making the method verbose.

Once features are recognized, a Feature Combination Generator based on Entity Growing Approach is used. This system analyzes boundary surfaces of features to derive interacting ones and transforms them into isolated features by expanding their material surfaces. The elaborated system can than handle both isolated and interacting features with the possibility of generating multiple interpretations of features from interacting ones. For more details on the previously developed AFR system, refer to Jaider et Al [3].



```
def process_cartesian_point(stepLine):
    cp_name = getLabelFromStepLine(stepLine)
    start = stepLine.find(',(')+2
    end = stepLine.find('))')
    params = stepLine[start:end].split(',')
    x = stepCode.REAL(params[0])
    y = stepCode.REAL(params[1])
    z = stepCode.REAL(params[2])
    cp = stepCode.cartesian_point(cp_name, [x,y,z])
    cp.stepLine = stepLine
    return cp
```

```
#57=CARTESIAN_POINT('cartesian point',(0,35.,-40))
```

Figure 4: Mapping Script Of Instance Parameters Of CARTESIAN_POINT Entity To Its Object Attributes

It has been found that this Feature Combination Generator can lead to combinational explosions when several features are interacting. For this reason, a system called Entity Suppressor has been developed to exclude interactions between some features, based on manufacturing rules and material removal principles. The system consists in adding or removing respectively to the part and from the stock, material rings matching to some volumetric features of the part. By this way, the part and the stock are modified, and the number of interacting features that are used to generate multiple interpretations of features is remarkably reduced, in other words, the part is simplified and interactions between the suppressed features and the remaining features are avoided. For more details, refer to [11].

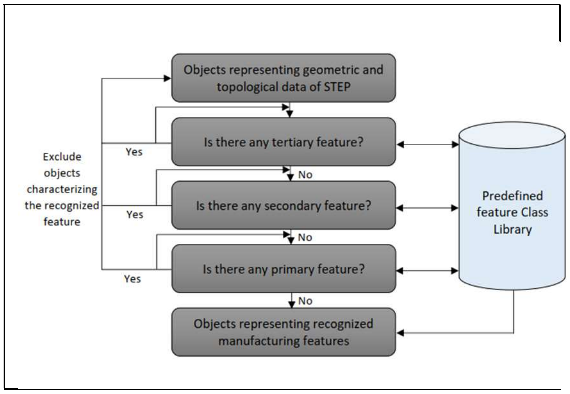


Figure 5: Automatic Feature Recognition Algorithm

5. CONVERSION OF THE RECOGNIZED MANUFACTURING FEATURES INTO TURNING MACHINING FEATURES DEFINED BY PART 12 OF ISO 14649

5.1 Translation of ISO 14649 (ARM) and ISO 10303-238 (AIM) EXPRESS schemas into Python classes

We have seen in paragraph 3-III that EXPRESS definitions of geometric and topological entities included in the AP203 application protocol schema have been translated into Python classes thanks to STEPcode library. In the same way, we have exploited this library to translate EXPRESS schema entities of ISO 14649 into Python classes (Figure 6). This EXPRESS schema brings together parts 10, 11, 12, 111, and 121 of ISO 14649 in a single file with the .exp extension. The EXPRESS schema on which our research is based on part 12 of ISO 14649, which describes all process data for turning.

It is important to note that this part 12 of ISO 14649 cannot be used alone, ISO 14649 does not contain definitions of geometric and topological AP203 entities. As a result, some attributes of machining features defined by part 12 of ISO 14649 are provided by ISO 10303-238, which includes AP203 entities. A good example is the attribute that concerns all the machining features, which is Feature_placement attribute. This attribute is described by the Axis2_placement_3d entity, which is required to define the position and orientation of the entity relative to the origin of the raw part. For this reason, the EXPRESS schema of ISO 10303-238 has also been translated into Python classes to complement the definitions of turning machining features. The EXPRESS schema of ISO 14649 (ARM) is available on the STEP Tools website [12]. Figure 6 shows the classes of some entities of ISO 10303-238 and ISO 14649. It should be noted that some machining features such as circular_face are not contained in the EXPRESS schema of ISO 14649, and therefore they are added manually as feature classes.

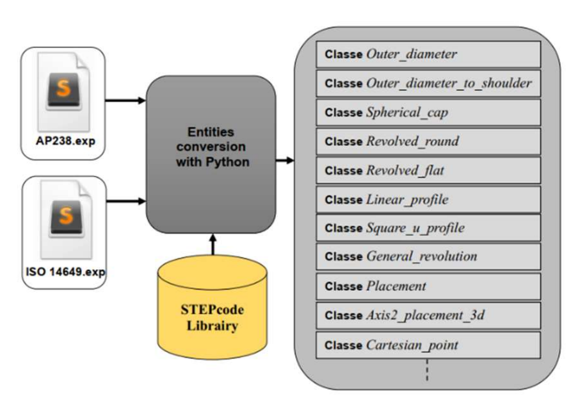


Figure 6: Translations Of The EXPRESS Schemas Of The AP283 And ISO 14649 Into Python Classes

5.2 Definition of geometric and dimensional attributes of the classes representing turning features of the ISO 14649 standard, according to parameters of the recognized machining features.

Taking the fact that all machining feature classes are obtained, only geometric and dimensional attributes of classes representing machining features of ISO 14649-12 will be defined in a first stage. Other attributes describing operations, strategies, working steps, the stock definition, etc, will be addressed in a future work. The definition of ISO 14649-12 turning features attributes is based on the extraction of geometric and dimensional data from recognized features, and the mapping of these data to ISO 14649-12 feature class attributes.

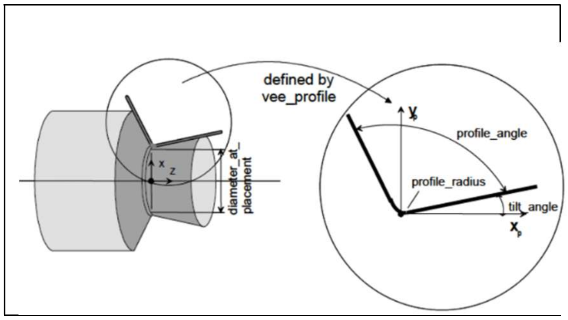


Figure 7: The Outer_Diameter_To_Shoulder Feature (ISO 14649-12)

It should be noted that when designing the raw cylinder and the finished part (which are used as input to our CAPP system) with a CAD modeler such as CATIA, their positions and orientations must be taken carefully into account in such a way to comply with ISO 14649-12 requirements. This means that the axis of the raw cylinder must be the Z axis, the right face of this cylinder must be superimposed with the origin (0) of the Z coordinate (this is the condition for positioning the cutting tool on the face of the raw cylinder in a CNC machine for the first machining operation which is facing), thus, the left face of the cylinder has a negative Z value. The final part must be designed separately and obviously be positioned to be enveloped by the raw cylinder, and having the same axis of revolution Z. Since in our work, any part designed is of a revolution type, and respects the requirements above, the optional attributes Axis and Ref_direction of all machining features can be represented respectively by the vectors V1 (0, 0, 1) and V2 (1, 0, 0). In this article, we have presented only one machining feature, since the methodology is the same for the other features.

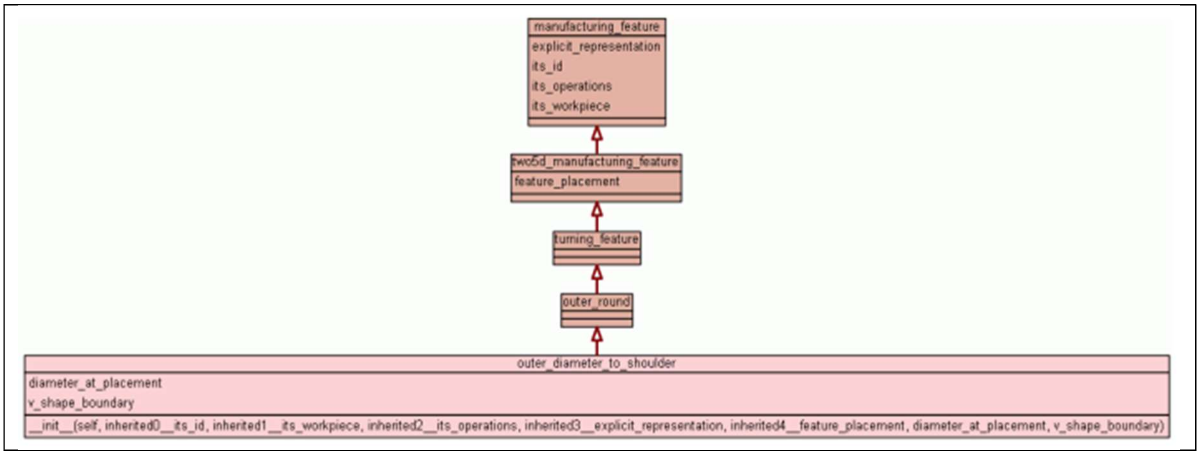


Figure 8: The Classe Outer_Diameter_To_Shoulder And Its Attributes

5.2.1 The Outer_diameter_to_shoulder machining feature

We assume that a shoulder feature can only be defined according to part 12 of ISO 14649 by the class Outer_diameter_to_shoulder. An Outer_diameter_to_shoulder is a type of outer_round that is a sweeping of a shape one complete revolution about an axis. The shape shall be specified by two lines that connect at a point and extend infinitely. The enclosed angle shall be smaller than 180° (Figure 7).

The Outer_diameter_to_shoulder class is defined by the attributes Diameter_at_placement and V_shape_boundary (Figure 8). The Diameter_at_placement attribute describes the diameter at the feature’s placement (co-ordinate system). The z co-ordinate is the position where the two sides of the vee_profile come together. (Figure 7). The V_shape_boundary attribute which is described by the vee_profile entity, specifies the revolved shape necessary to define the Outer_diameter_to_shoulder machining feature. The placement of the profile shall be along the x-axis of the outer_diameter_to_shoulder at a specified distance away from the origin. The orientation of the y-axis of the vee_profile shall be the same as the x-axis of the outer_diameter_to_shoulder and the x-axis of the vee_profile shall be the same as the z-axis of the outer_diameter_to_shoulder (Figure 7). Vee_profile is described by the attributes; Profile_radius, Profile_angle, Tilt_angle, and attributes added by ISO 10303-238; First_side_length and Second_side_length (Figure 9), as well as the optional attribute Placement, which describes the location of the profile relative to the feature, and which is inherited from the class Profile. If this attribute is omitted, the origin of the profile will be a point having the same z coordinate with the origin of the coordinate system of the feature. The first_side_length indicates the distance, as measured from the profile origin, along the side of the vee located by the tilt_angle parameter. The second_side_length indicates the distance, as measured from the profile origin, along the side of the vee located by the sum of the tilt_angle and profile_angle.

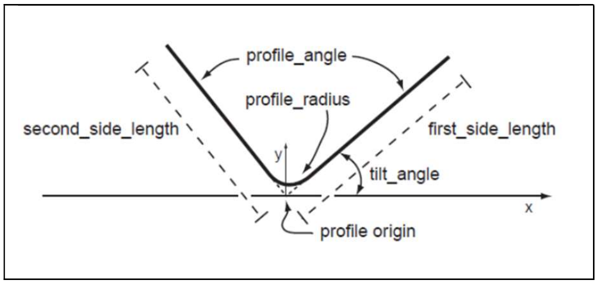


Figure 9: Parameters Of The Classe Vee_Profile According To ISO 10303-238 [13]

After presenting all geometric and dimensional parameters defining the Outer_diameter_to_shoulder class, the next step is to map parameters of the recognized shoulder feature to geometric and dimensional attributes of the Outer_diameter_to_shoulder class. Recalling that the recognized feature retains all its geometric and topological attributes, such as the edges which delimit its loops, the radii and centers of circles, the coordinates of the vertices, etc. Figure 10 brings together the different parameters necessary to instantiate a new Outer_diameter_to_shoulder object, according to part 12 of ISO14649. According to Han [14], the milling cutter is of infinitesimal radius, and therefore a profile with convex vertices is considered acceptable. Hence, in turning, we assume that the insert nose radius is of infinitesimal radius, and therefore, the radius of the connected lines of the shoulder profile is acceptable to be zero. Hence, the Profile_radius attribute is assumed to be null. For simplification, we also assume that a shoulder is formed by the junction of a cylindrical surface and a conical surface along a circular edge loop A (Figure 10), and therefore, the attribute Tilt_angle can take a null value. Thus, the diameter D is equal to the diameter of a semicircle which forms this circular edge loop A. The origin O of the coordinate system of the feature Outer_diameter_to_shoulder is positioned on the axis of rotation. The z coordinate of this origin O is equal to the z coordinate of the center of one of the semicircles which form the edge loop A. The location (o) of the origin of the profile is obtained by the intersection of the circular edge loop A with the half-plane xz (x positive). Thus, the directions z and x of the profile will be defined by the vectors V1 (0,1,0) and V2 (0,0,1).

In STEP, a cone is defined by a radius and the angle between the axis of rotation of the conical surface and a line of the cone contained in the same plane, for example in the XZ plane (Figure 11). An example of a cone with angle α of 30 ° and radius of 15mm is given in STEP by the following record: # 40 = CONICAL_SURFACE ('Cone', # 39,15,0.523598775598). Note that the value 0.523598775598 of the angle of the cone is in radian (equivalent to 30 °). Hence, the angle defined by the Profile_angle attribute (in degrees) is equal to the difference between an angle of 180 ° and the angle of the cone.

The remaining parameters to extract are First_side_length and Second_side_length. The attribute First_side_length defines the length L1 which can be calculated by the difference (in absolute value) between the z coordinates of the semicircles centers (O and B in Figure 10) of one of the edge loops which delimit the cylinder. The Second_side_length attribute defines a length that can be calculated using the cone parameters. The length L can be calculated by the difference (in absolute value) between the z coordinates of the semicircles centers (O and C in Figure 10) of one of the edge loops which delimit the cone. Therefore, L2 is equal to L devised by the cos (α).

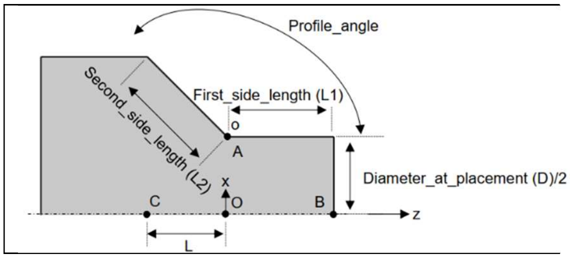


Figure 10: Geometric And Dimensional Parameters Defining The Feature Outer_Diameter_To_Shoulder

5.2.2 Modification of geometric and dimensional parameters of design features

In many cases, design features of the final part geometry can be used directly for process planning, supplemented with attributes such as required technology, tools, operations, and strategies... However, depending on the technology used, the planned operations and their sequencing, interactions between features, improvement of surface quality, additional machining features must be created at the level of the CAM system [13], and / or volumes of design features must be modified [3, 11, 15, 16, 17].

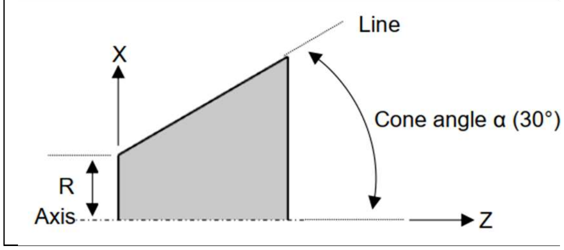


Figure 11: Definition Of The Angle Of A Cone In STEP

In ISO 14649, machining features are used to describe the volumes to be removed by machining to obtain the final shape of the part, which is described by the design features in ISO 10303-224 and ISO 10303-214. AP224 features are recognized by a CAD, CAM, or a feature recognition system, which identify features of the final shape of the part. However, these features are not necessarily the same features of ISO14649. The features of ISO14649 which are based on the geometry of the blank and the final geometry derived from design features, can be modified, i.e. in order to solve the problem of interactions between features. Machining features of ISO 14649 are closely related to the machining process defined by the process plan. If a manufacturing engineer creates another work plan to get the same part, the new process plan can use a completely different set of machining features (defined by ISO 14649). Figure 12 shows how machining features volumes derived from design features can be obtained. These features are produced with different machining operations (facing and longitudinal turning) to produce the same part. Both process plans produce the same result, but each calls for different features.

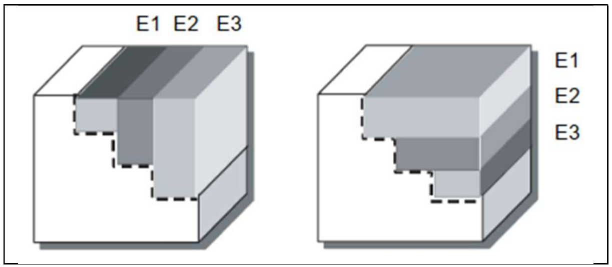


Figure 12: Alternative machining operations and features (ISO 10303-238)

As we will see in this section, due to interactions between features, dimensional parameters of certain features such as cylinders and faces must be modified. If we look at Figure 13, many combinations of features are generated to produce the same part. Consequently, certain dimensional modifications of some design features must be done to correctly define new volumes of machining features which will be defined according to ISO 14649. At this stage, it should be noted that dimensional attributes of design features must imperatively be modified according to those of a candidate combination of features. As a first step in this work, we have frozen in our system to define new volumes of machining features according to the frequent combination 1 of Figure 13, by the fact that it respects an economic constraint which consists in reorganizing machining passes for a minimum machining time. Therefore, dimensional parameters of faces, cylinders and shoulders must be modified.

Modification of face feature parameters

As explained in paragraph 4.2 of the paper [11], given that the face is the first feature to be machined, it must be extended radially until reaching the cylindrical surface of the stock, and therefore, its new diameter defined by Diameter_at_placement attribute becomes equal to the diameter of the raw cylinder (Figure 14).

Modification of outer_diameter (cylinder) feature parameters

Since the cylinder with the largest diameter of the part is extended after having extended the face (if the cylinder is not adjacent to the face), its new length defined by the attribute Feature_length becomes equal to the difference (in absolute value) between the z coordinate of the center (A) of one of the left semicircles of a loop delimiting the cylinder, and the z coordinate of the center (B) of one of the semicircles which delimit the planar face. The origin placement of the cylinder feature coordinate system defined by the Location attribute becomes superimposed with the center (B).

Modification of Outer_diameter_to_shoulder feature parameters

To comply with combination 1 of Figure 13, only the cylindrical surfaces (including those being part of shoulders) of the part must be extended. Since the face is extended in the first place, all the cylinders that are part of shoulders and which are not adjacent to the face will be extended until reaching this face (case of the cylinder part of the shoulder F3 in Figure 14). Hence, the new length defined by the attribute First_side_length (L1 in Figure 14) becomes equal to the difference (in absolute value) between the z coordinate of the center O of one of the semicircles on the left edge loop delimiting the cylinder, and the z coordinate of the center B of one of the semicircles which delimit the planar face. The location of the shoulder remains unchanged.

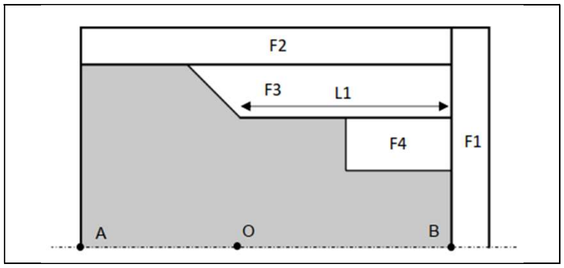
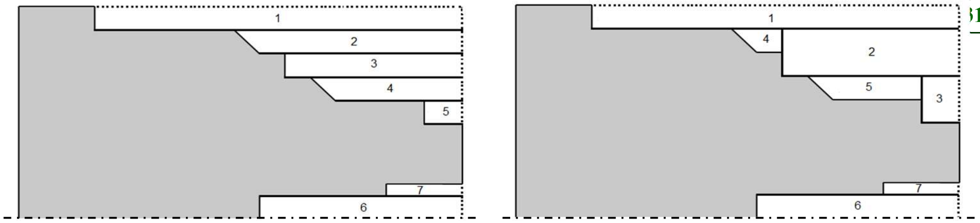
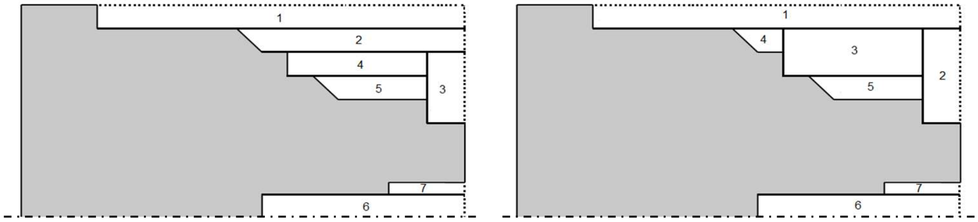


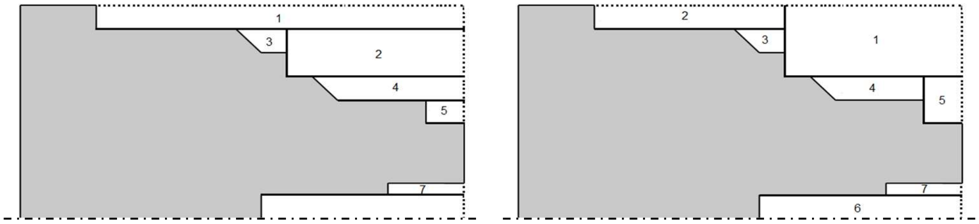
Figure 14 : Dimensional parameter modifications of features ; face, cylinder, shoulder



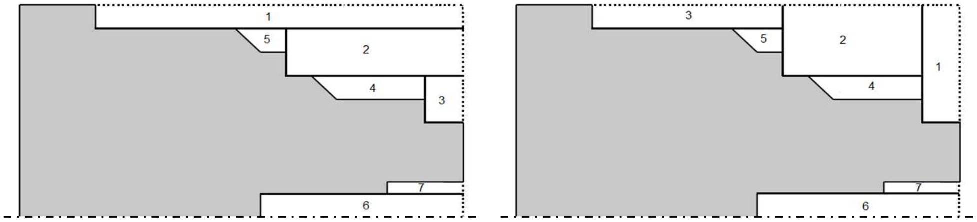
Combination 1 Combination 2



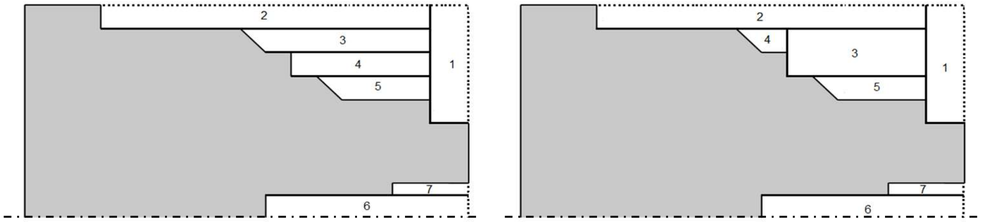
Combination 3 Combination 4



Combination 5 Combination 6



Combination 7 Combination 8



Combination 9 Combination 10

Figure 13 : Some Features Combinations To Machine The Same Part

6. CASE STUDY : TEST PART

The example part of Figure 15 was taken in this paragraph to show in one hand, how the developed CAPP-Turn system works, how geometric and topological data of the part are extracted and reorganized in a readable manner, and on the other hand, to illustrate the results of feature recognition

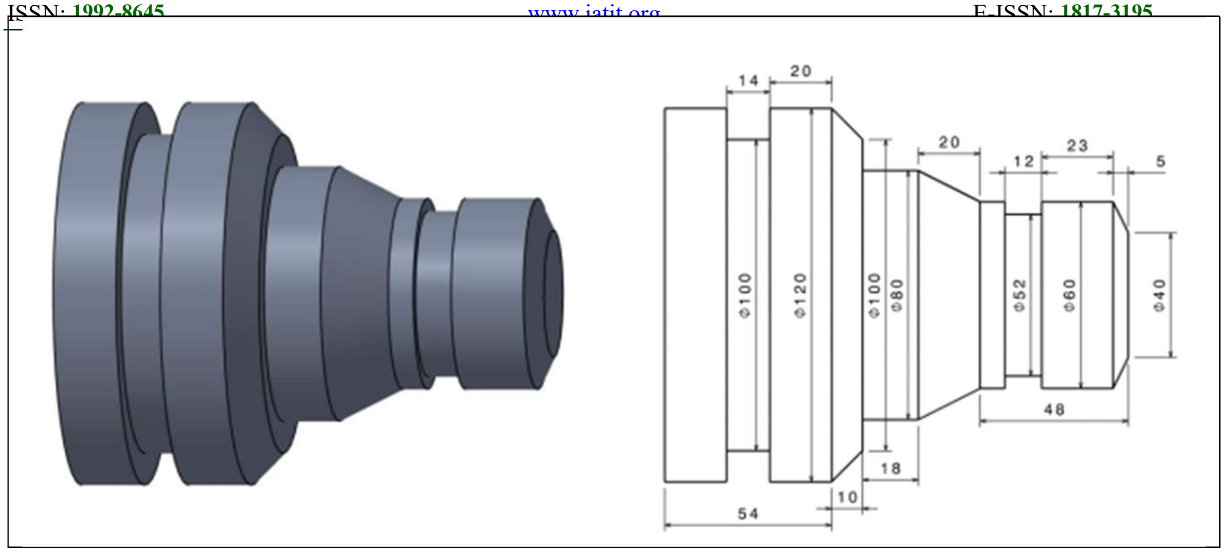


Figure 15 : Example Part

and more particularly to visualize parameters of the recognized shoulder feature explained before.

6.1 Selection of the part and the raw cylinder: Case study

Before starting the processing of a part, the process planner must obviously have CAD models (STEP files) of the finished part and at least one raw cylinder. The process planner can import STEP file of the part using the button "import part STEP file". Once the part STEP file has been entered into the system, the geometry can be displayed by default in a 3D graphic frame on the right side of the screen, called "3D View (Workpiece) " (Figure 16). Note that the OCC library used in CAPP-Turn to display the part only supports STEP AP203.

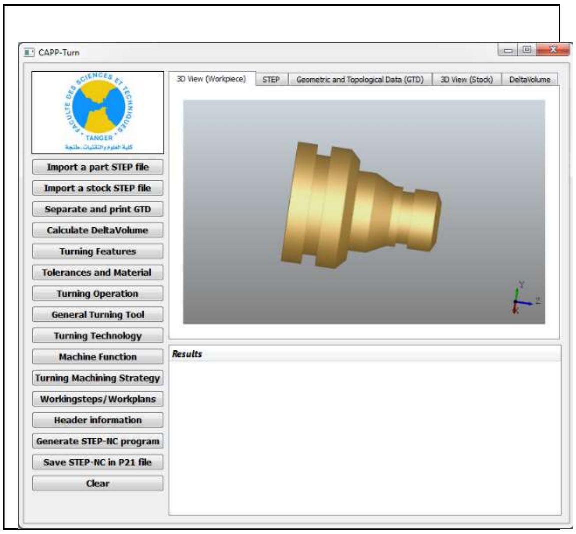
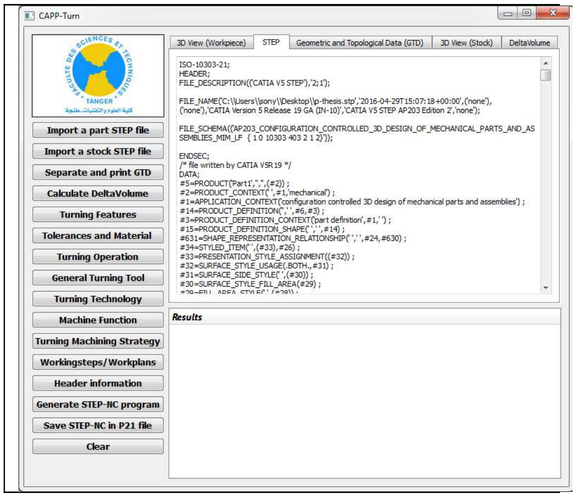


Figure 16: 3D Model Of The Processed Part

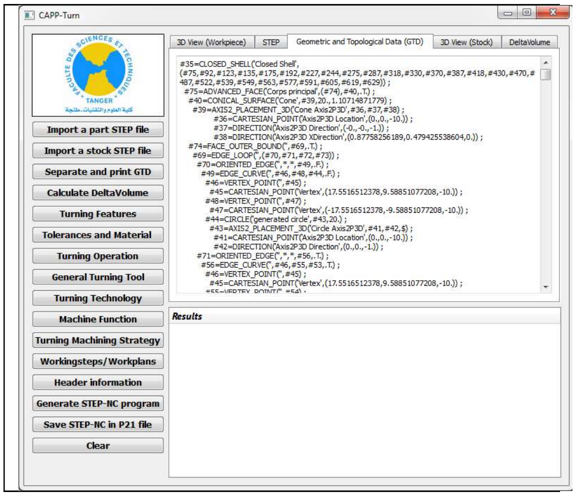
The raw content of the part STEP file can be displayed in a text area by clicking on the "STEP" button (Figure 17).



```
ISO-10303-21;
HEADER;
FILE_DESCRIPTION(('CATIA V5 STEP'),'2;1');

FILE_NAME('C:\\Users\\sony\\Desktop\\p-thesis.stp','2016-04-29T15:07:18+00:00',('none'),
('none'),'CATIA Version 5 Release 19 GA (IN-10)','CATIA V5 STEP AP203 Edition 2','none');

FILE_SCHEMA(('AP203_CONFIGURATION_CONTROLLED_3D_DESIGN_OF_MECHANICAL_PARTS_AND_AS
SEMBLIES_MIM_LF  { 1 0 10303 403 2 1 2}'));

ENDSEC;
/* file written by CATIA V5R19 */
DATA;
#5=PRODUCT('Part1','','',(#2)) ;
#2=PRODUCT_CONTEXT(' ',#1,'mechanical') ;
#1=APPLICATION_CONTEXT('configuration controlled 3D design of mechanical parts and assemblies') ;
#14=PRODUCT_DEFINITION('',' ',#6,#3) ;
#3=PRODUCT_DEFINITION_CONTEXT('part definition',#1,' ') ;
#15=PRODUCT_DEFINITION_SHAPE(' ',' ',#14) ;
#631=SHAPE_REPRESENTATION_RELATIONSHIP(' ',' ',#24,#630) ;
#34=STYLED_ITEM(' ',(#33),#26) ;
#33=PRESENTATION_STYLE_ASSIGNMENT((#32)) ;
#32=SURFACE_STYLE_USAGE(.BOTH.,#31) ;
#31=SURFACE_SIDE_STYLE(' ',(#30)) ;
#30=SURFACE_STYLE_FILL_AREA(#29) ;
```

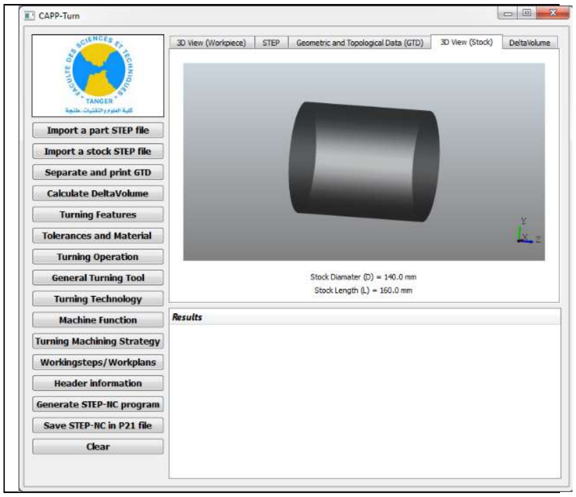
Figure 17: Raw Content Of The Part STEP File

By clicking on the button "Separate and print GTD", parameters of entity instances contained in the part STEP file are mapped to objects attributes of geometric and topological entity classes that are based on the EXPRESS schema of STEP AP203. Since reading the raw STEP file is difficult, we thought of ordering automatically instances of the STEP file starting with the highest-level entity instance (CLOSED_SHELL), thus, instances of geometric and topological entity instances related to each face (ADVANCED_FACE) are displayed in a tree structure (Figure 18), making easier for the user to read the STEP data file. In the same way as the finished part, the STEP file of a raw cylinder can be imported by the button "import a stock STEP file". The 3D geometry of the raw cylinder can be displayed in a 3D graphic frame on the right side of the screen by clicking on the "3D View (Stock)" button (Figure 19). By clicking on the "Calculate Delta Volume" button, the user can display the Boolean subtraction of the raw cylinder and the part in a 3D graphics frame "Delta Volume" (Figure 20). This subtraction represents the removable volume of material.



```
#35=CLOSED_SHELL('Closed Shell',
(#75,#92,#123,#135,#175,#192,#227,#244,#275,#287,#318,#330,#370,#387,#418,#430,#470,#
487,#522,#539,#549,#563,#577,#591,#605,#619,#629)) ;
 #75=ADVANCED_FACE('Corps principal',(#74),#40,.T.) ;
  #40=CONICAL_SURFACE('Cone',#39,20.,1.10714871779) ;
   #39=AXIS2_PLACEMENT_3D('Cone Axis2P3D',#36,#37,#38) ;
        #36=CARTESIAN_POINT('Axis2P3D Location',(0.,0.,-10.)) ;
        #37=DIRECTION('Axis2P3D Direction',(-0.,-0.,-1.)) ;
        #38=DIRECTION('Axis2P3D XDirection',(0.87758256189,0.479425538604,0.)) ;
  #74=FACE_OUTER_BOUND('',#69,.T.) ;
   #69=EDGE_LOOP('',(#70,#71,#72,#73)) ;
    #70=ORIENTED_EDGE('',*,*,#49,.F.) ;
     #49=EDGE_CURVE('',#46,#48,#44,.F.) ;
      #46=VERTEX_POINT('',#45) ;
       #45=CARTESIAN_POINT('Vertex',(17.5516512378,9.5885107720 8,-10.)) ;
      #48=VERTEX_POINT('',#47) ;
       #47=CARTESIAN_POINT('Vertex',(-17.5516512378,-9.5885107720 8,-10.)) ;
      #44=CIRCLE('generated circle',#43,20.) ;
       #43=AXIS2_PLACEMENT_3D('Circle Axis2P3D',#41,#42,$) ;
        #41=CARTESIAN_POINT('Axis2P3D Location',(0.,0.,-10.)) ;
        #42=DIRECTION('Axis2P3D Direction',(0.,0.,-1.)) ;
    #71=ORIENTED_EDGE('',*,*,#56,.T.) ;
     #56=EDGE_CURVE('',#46,#55,#53,.T.) ;
      #46=VERTEX_POINT('',#45) ;
       #45=CARTESIAN_POINT('Vertex',(17.5516512378,9.5885107720 8,-10.)) ;
```

Figure 18: Tree Structure Of Entity Instances Of The Part STEP File



Stock Diamater (D) = 140.0 mm
Stock Length (L) = 160.0 mm

Figure 19: The Raw Cylinder And Its Parameters (D, L)

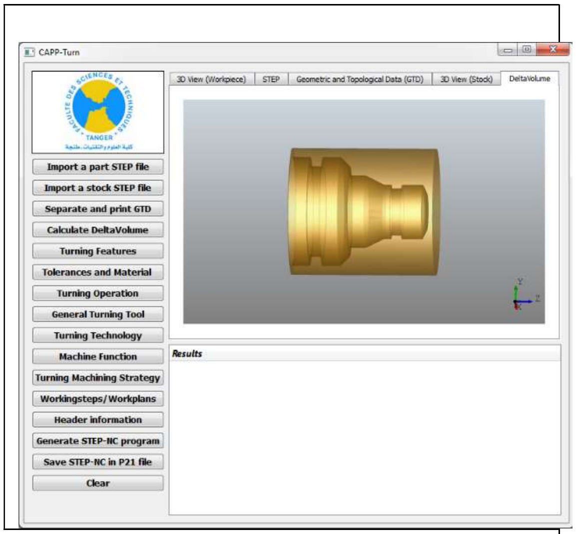
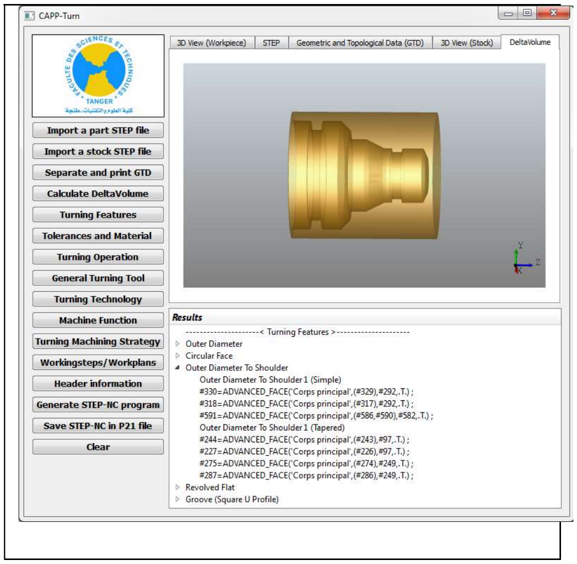


Figure 20: Delta Volume

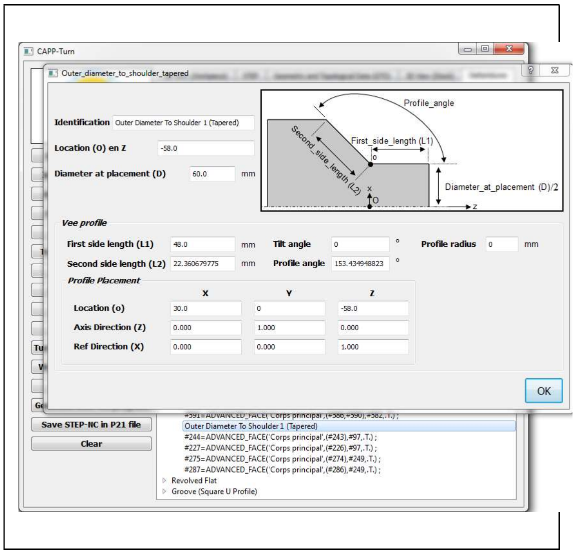


```
---------------------< Turning Features >---------------------
▷ Outer Diameter
▷ Circular Face
◢ Outer Diameter To Shoulder
     Outer Diameter To Shoulder 1 (Simple)
     #330=ADVANCED_FACE('Corps principal',(#329),#292,.T.) ;
     #318=ADVANCED_FACE('Corps principal',(#317),#292,.T.) ;
     #591=ADVANCED_FACE('Corps principal',(#586,#590),#582,.T.) ;
     Outer Diameter To Shoulder 1 (Tapered)
     #244=ADVANCED_FACE('Corps principal',(#243),#97,.T.) ;
     #227=ADVANCED_FACE('Corps principal',(#226),#97,.T.) ;
     #275=ADVANCED_FACE('Corps principal',(#274),#249,.T.) ;
     #287=ADVANCED_FACE('Corps principal',(#286),#249,.T.) ;
▷ Revolved Flat
▷ Groove (Square U Profile)
```

Figure 21: The Recognized Machining Features Of The Part

6.2 Automatic feature recognition

Taking the fact that STEP files of the raw cylinder and the finished part are entered into the system, automatic feature recognition can be carried out simply by clicking on the button "Turning Features". In the Results section, the recognized machining features are displayed in a tree (Figure 21). By clicking on the small triangle next to a feature, features of a same type are displayed with the faces (ADVANCED_FACEs) that compose them. To display parameters of a feature, simply double-click on its name in the feature tree, a dialog box containing parameters of the features can be displayed. Figure 22 shows parameters of a recognized shoulder. Figure 23 shows an extract from the STEP-NC file generated by our CAPP-Turn application, which contains instances of geometric and dimensional entities of the recognized shoulder.



Identification Outer Diameter To Shoulder 1 (Tapered)
Location (O) en Z -58.0
Diameter at placement (D) 60.0 mm

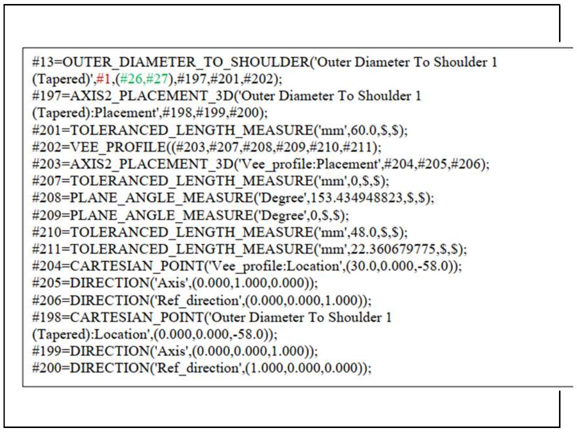
Vee profile
First side length (L1) 48.0 mm **Tilt angle** 0 ° **Profile radius** 0 mm
Second side length (L2) 22.360679775 mm **Profile angle** 153.434948823 °

Profile Placement

	X	Y	Z
Location (o)	30.0	0	-58.0
Axis Direction (Z)	0.000	1.000	0.000
Ref Direction (X)	0.000	0.000	1.000

```
#591=ADVANCED_FACE('Corps principal',(#586,#590),#582,.T.);
Outer Diameter To Shoulder 1 (Tapered)
#244=ADVANCED_FACE('Corps principal',(#243),#97,.T.);
#227=ADVANCED_FACE('Corps principal',(#226),#97,.T.);
#275=ADVANCED_FACE('Corps principal',(#274),#249,.T.);
#287=ADVANCED_FACE('Corps principal',(#286),#249,.T.);
Revolved Flat
Groove (Square U Profile)
```

Figure 22: Parameters Of The Recognized Shoulder Feature



```
#13=OUTER_DIAMETER_TO_SHOULDER('Outer Diameter To Shoulder 1
(Tapered)',#1,(#26,#27),#197,#201,#202);
#197=AXIS2_PLACEMENT_3D('Outer Diameter To Shoulder 1
(Tapered):Placement',#198,#199,#200);
#201=TOLERANCED_LENGTH_MEASURE('mm',60.0,$,$);
#202=VEE_PROFILE((#203,#207,#208,#209,#210,#211);
#203=AXIS2_PLACEMENT_3D('Vee_profile:Placement',#204,#205,#206);
#207=TOLERANCED_LENGTH_MEASURE('mm',0,$,$);
#208=PLANE_ANGLE_MEASURE('Degree',153.434948823,$,$);
#209=PLANE_ANGLE_MEASURE('Degree',0,$,$);
#210=TOLERANCED_LENGTH_MEASURE('mm',48.0,$,$);
#211=TOLERANCED_LENGTH_MEASURE('mm',22.360679775,$,$);
#204=CARTESIAN_POINT('Vee_profile:Location',(30.0,0.000,-58.0));
#205=DIRECTION('Axis',(0.000,1.000,0.000));
#206=DIRECTION('Ref_direction',(0.000,0.000,1.000));
#198=CARTESIAN_POINT('Outer Diameter To Shoulder 1
(Tapered):Location',(0.000,0.000,-58.0));
#199=DIRECTION('Axis',(0.000,0.000,1.000));
#200=DIRECTION('Ref_direction',(1.000,0.000,0.000));
```

Figure 23: red = WORPIECE (raw cylinder), Green = Machining Operations, Black = Geometric And Dimensional Instances

7. CONCLUSION

In this paper, an upgrade of the method for extracting geometric and topological parameters of a STEP AP203 part file has been presented. At the beginning, the EXPRESS schema of the AP203 is parsed via the STEPCode library to obtain geometrical and topological entity classes. Mapping scripts have been developed to extract geometric and topological parameters from entity instances contained in the AP203.exp file. These parameters are used by the entity classes obtained, to create objects that represent geometric and topological data of a part.

By analyzing these geometric and topological objects, the rule-based system identifies each manufacturing feature according to certain pre-established rules, which are characteristic to the feature. Once features are recognized, the system adopting the Entity Growing approach generates a set of features according to a selected combination, by extending their material surfaces.

The new geometric and dimensional parameters of the volumetrically modified machining features are finally used to define just geometric and dimensional parameters of turning machining features defined according to ISO 14649-12. Definition of other attributes describing machining operations, cutting tools, working steps, etc, will be the subject of a future work.

REFERENCES:

[1] Chen Wong Keong, Yusri Yusof, "Developing a Feature-based System for Automated Machining Feature Recognition (ISO 10303 AP 224) of Prismatic Components", Applied Mechanics and Materials Vols 229-231 (2012) pp2375-2379.

[2] Tahir Fidan, Saleh AMAITIK, S. Engin Kilic, "Concept of STEP AP224 features based modeling for rotational parts" Conference Paper · January 2003.

[3] Jaider Oussama, Elmesbahi Abdelilah, Rechia Ahmed, "Manufacturing Computer Aided Process Planning For Rotational Parts. Part 1: Automatic Feature Recognition From STEP AP203 Ed2," Int. Journal of Engineering Research and Applications, ISSN: 2248-9622, Vol. 4, Issue 5 (Version 1), May 2014, pp.

[4] Shah Jami. J et Mantyla Martti, "Parametric and Feature Based CAD/Cam: Concepts, Techniques, and Applications", Book, - New York, NY, USA : John Wiley & Sons, Inc., 1995.

[5] T.Fidan, S.M. Amaitik, E. S. Kiliç, "Concept of step ap224 features based modeling for rotational parts", 3rd International Conference

on Research and development in mechanical industry, RaDMI 2003.

[6] ISO 10303-224:2006, Industrial automation systems and integration — Product data representation and exchange — Part 224: Application protocol: Mechanical product definition for process planning using machining features.

[7] M. Kang, J. Han, J.G. Moon, "An approach for interlinking design and process planning", Journal of Materials Processing Technology 139 (2003) 589–595.

[8] Chen Wong Keong, "Design A System and Database for Automated Machining Feature Recognition", Prosiding 037, CiETVET2012.

[9] http://stepcode.org/

[10] https://github.com/stepcode/stepcode/wiki

[11] Oussama Jaider, Abdelilah El Mesbahi, and Ahmed Rechia, "Manufacturing Computer Aided Process Planning For Rotational Parts. Part 2: A New Approach for Optimizing Multiple Interpretations of Interacting Features Based on Manufacturing Rules and Metal Removal Principals," International Journal of Innovation and Applied Studies, vol. 8, no. 1, pp. 307–319, September 2014.

[12] http://www.steptools.com/library/stepnc/tech_resources/stepnc_arm_merged.exp

[13] ISO, ISO 10303-238: Industrial automation systems and integration – Product data representation and exchange - Part 238: Application Protocols: Application interpreted model for computerized numerical controllers. 2007.

[14] J. Han, "3D Geometric Reasoning Algorithms for Feature Recognition", PhD thesis, Computer Science Department, University of Southern California (USC), 1996.

[15] Ersan ASLAN, "Process unification and frame preparation of machining parameters for rotational parts," Journal of Engineering Sciences, SAYFA: 137-145 (2005).

[16] Yakup Yildiz, İhsan KOrkut, Ulvi Şeker, "Development of a Feature Based CAM System for Rotational Parts", G.U. Journal of Science, 19(1): 35-40 (2006).

[17] Md. Deloyer Jahan, Golam Kabir, "Development of Computer Aided Process Planning (CAPP) for Rotational Parts", Global Journal of Researches in Engineering, Page 16 Vol.10 Issue 4 (Ver 1.0), September 2010.