Proving manufacturability at the design stage using commercial modeling Software: Through feature mapping and feature Accessibility

S. Srikumaran[1] and S. Sivaloganathan[2]

[1]Brunel University, srikumaran30@yhaoo.co.uk
[2]Brunel University, s.siva@brunel.ac.uk

ABSTRACT

It is a fundamental requirement that during the design of a component, its manufacturability to the required quality at an acceptable cost must be ensured. In machining, achieving the required quality is ensured by performing all machining quality critical operations in a single setting-up. Paper outlines the basis for the manufacturability of a component to the required quality and cost while it is at the design stage through feature mapping. A comprehensive analysis of features in a feature based design system and manufacturing process (CNC milling) formed the basis of the mapping sequence to link the both classes of features. Once the manufacturing features of the component in question are mapped ensuring its accessibility and number of setting required can pave the way for ensuring that the above requirement is met. Design features whose geometry is difficult to machine and/or require many settings affecting quality require modification or reorientation. In essence introduces a tool for identifying the machining features present in a design feature model for individual machining set up.

Keywords: Design Feature, Manufacturing Feature, Feature Mapping, Set up Planning, Pro/Engineer, Pro/Toolkit.

1. INTRODUCTION

The current frontiers of knowledge that is relevant to ensuring the manufacturability of a component to the required quality and cost while it is at the design stage has seen many changes. In particular (a) the development of 'Solid Modelling' from the early stages to geometry-based systems that were developed into a high level Feature Based Modelling System (b) Feature Recognition that permits the derivation of the model using a particular class of features from a model made up of another class of features or of basic geometry (c) the developments and definitions of Manufacturing Features and (d) application programmers interface which permits the access of parameters that are necessary for establishing a Manufacturing Feature Model from a Design Feature Model of an object.

Features can be defined as the building blocks of solid models. Shah [3] defines features as "Stereotypical knowledge structures embedded in cognitive process in design, analysis, planning, and all other engineering activities and that features are necessarily view point and application dependent". Features are elements at a semantically higher level than pure geometric elements typically used in CAD/CAM systems Mäntylä[4]. Features represent the engineering meaning of the geometry of a part or assembly. The meaning may involve the formalization of the functions the feature serves or how it can be produced. Features are parametric shapes associated with attributes as in its intrinsic geometric parameters, length, width, and depth, as well as position, orientation, geometric tolerances, material properties and references to other features. Features are application specific and they support multiple viewpoints e.g. design, process planning and assembly Chang[5]. Feature definitions in Contemporary Feature Based Design systems build any complex geometry shape as an assemblage of predefined feature instances. Instantiated features incorporate their generic and specific properties.

The two main concepts commonly used in feature-based modelling are: Design by Features and Feature Recognition. Design by Features provides a library of pre-defined features to the user. Feature Recognition is the derivation of the 'Feature Model' from a 'Geometric Model' or geometric information or from a solid model based on different feature class library. Automatic feature recognition has been approached in different ways. They are: Graph Based (Joshi[6], Gavankar[7], De Floriani's [8,9], Marefat [10], Trika's [11]), Volumetric based (Bernard[12], Tang[13], Kim[14,], Sakurai[17]), Rule based and Evidence based reasoning methods (Vandenbrande[18], Regli [21]) and Syntactic pattern recognition (Jakubowski [22]).

Application specific machining feature classification has many different approaches but sharing similarities. A manufacturing feature is defined as " collection of related geometric elements which as a whole correspond to a particular manufacturing method or process. They can be used to reason about the suitable manufacturing process or methods to create the geometry" [3]. Satyandra[23] define machining feature as "the portion of work piece affected by machining operation". Kramer[24] developed a library of Material Removal Shape Element Volumes (MRSEV's) as a means of categorizing the shape volumes to be removed by machining operation on a 3-axis machining centre. Generic approach in defining a machining feature remained as "the volume of material that can be removed by machining". Joshi et.al[6] define machining features as "regions of a part having some manufacturing significance in the context of machining such as hole, slot pocket etc". Expert Machinist the feature based milling scheme defines machining-specific features as "material to be removed by 2-1/2 axis production milling based upon "shop floor" terminology and that are independent of tool path creation". These features define the material to be removed, based upon machining geometric shapes such as pocket, slot, profile etc and are independent of tool path creation.

The strategy adopted in proving the manufacturability of a feature-based design is, a mapping is carried out between the design features and the given set of manufacturing features. If any design feature fails in the mapping process (i.e.) cannot be mapped with in the given set of manufacturing features, one way forward is to identify the feature, which causes the difficulty, and try and design that feature out. Alternatively different manufacturing process has to be investigated. Once the design is mapped if the set of manufacturing features were for machining the accessibility of features for each setting is evaluated. Attention has to be paid to features that require higher precision and place them so that they can be machined in a single setting and also features that demand additional setting or fixtures can be re oriented. In the application of the methodology is implemented through feature based Commercial software Pro/Engineer, using Machining features defined in expert machinist and application programmer interface Pro/Toolkit.

2. APPROACH

The first feature class that is used by contemporary CAE systems is the 'Design Feature'. Design feature class permits a user to define an object easily and precisely using their class members. However their use is limited only to visualisation. The data contained in them should be transformed into other Feature Classes that represent other applications such as machining, inspection, tolerancing etc. This mapping of Design Features into Machining Features in a way suitable for a designer's use in checking their machinability and accessibility while machining is the problem addressed by this paper.

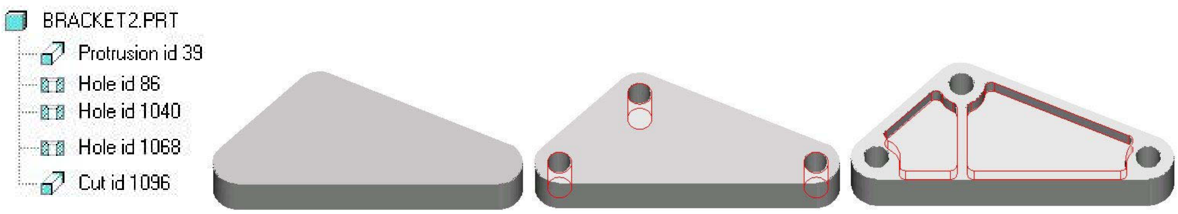
Fig. 1 explains the hierarchy of the modelling systems and their constituent elements used in building a CAD model. The bottom layer is Geometry Based Systems which are built with lower level geometry entities such as points, lines and curves, planes and solid lumps. Then in the intermediate layer the Feature Based Designs which are built using features that are blocks of geometry based entities are placed. In the top layer is the Mapped Systems, which are built with group of basic features used at the intermediate layer. This top layer is used as the core in addressing the manufacturability of a product.

High Level feature Mapping	Manufacturing Features as a Combination of Design Features	New Level of Abstraction
Feature Based Design	Solids Bounded By Design Features	Level of Abstraction with FBD
Geometry Based System	Low Level geometry and Topology	

Fig. 1. Solid modeling techniques and their level of abstraction.

2.1. Feature Based Design

The Features are shape representations with a functional requirement that are used in modelling application (views such as design, manufacture, inspection) specific products. Pro/Engineer is a commercial feature based design system. Fig. 2 shows the instancing of generic features in creating a part model (of a bracket) with its assemblage of feature instances. The feature tree (Fig. 2(a)) helps in identifying the individual features that make up the part and the sequence in which they are added. It consists of a base feature (protrusion), three holes and a cut feature. Thus the part is represented as an assemblage of five solid features using Pro/Engineer.



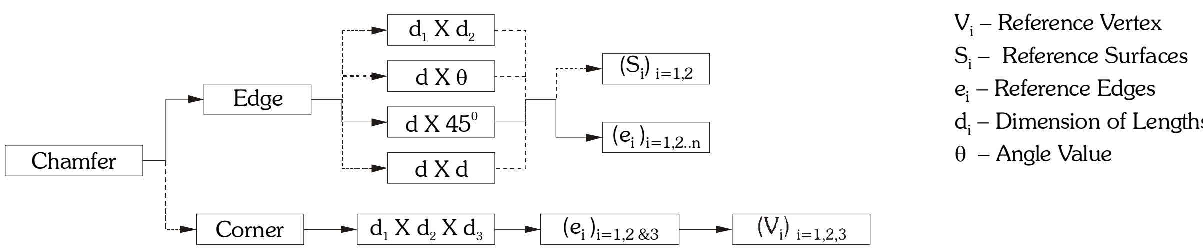
Protrusion id 39 Hole Id's 86,1040 &1068 Cut id 1096

(a) Feature Tree (b) Feature addition sequences

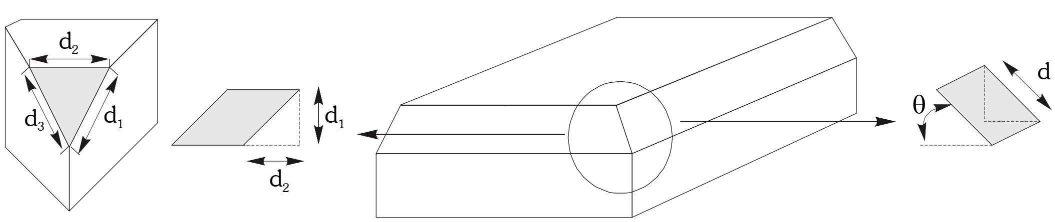
Fig. 2. The Bracket Part as an assemblage of design features.

In feature modeling a set of generic features defined in the feature library are instanced with specific properties and values in creating the application oriented product. Each generic feature definition has topology, set of parameters and constraints that are determined by the user when instancing these features during the modeling process. The validity of features is specified in generic feature definition by means of semantic constraints and if any of the constraints are violated in instancing the features the model fails.

With in the given set of feature creation options the designer has the freedom to select the different options and numerical values in creating his design feature. The fundamental Geometric shape together with other constraints such as placement position with respect to references, orientation, dimensional values form the basic characteristics of a feature modeled. Fig. 3 gives the steps involved in creating a chamfer feature. Which ever the branch chosen in creating the chamfer feature the corresponding information is stored in the feature model that can be retrieved and used further downward applications.



(a) Different options and their respective parameters and references available in Chamfer Feature creation.



(b) Corner feature (c) chamfer Features

Fig. 3. Instancing of chamfer feature.

Design by features is a very efficient method of creating a solid model with engineering meaning. However it will by definition have only a single application, the one for which the defining feature class was intended for. In order to make it more susceptible for several applications one should be able to derive different feature models that are different application oriented from a particular feature model

2.2 Feature Recognition and Mapping

Feature recognition is a critical interface for a variety of down stream applications. In this paper it is tightly coupled with manufacturing process planning. Advances in solid modelling representations have not produced a corresponding advance in feature recognition techniques. Motivation for feature recognition includes parts classification for group technology, generation of tool paths for CNC machines, and creation of process plans for part manufacture [8]. Feature recognition has been approached using variety of techniques, some of which are easier to parallelise than others. The approach in linking CAD and CAM is to use the CAD database and geometric modellers for design and making a recognition process that's application specific, on the basis of domain dependent feature recognition methods.

If the design feature does not correspond exactly to the process oriented features, a transformation process called mapping is required. Shah et. al [3] defines mapping classes when transforming feature models between different spaces. Feature mapping like feature recognition, requires geometric reasoning to find features of interest to a given application. Features in any domain are used to package certain topological and geometric relations and entities that reflect that domain's point of view. Geometric reasoning and/or computations can be used to derive application specific features. Theoretically the problem can be solved by storing a superset of relationship, from which all view point models can be derived by selectively extracting and grouping relationships and entities of interest.

2.3 Manufacturing Features in Expert machinist

Expert Machinist is a feature based manufacturing system that streamlines the NC programming process for 2-1/2-axis production milling. It defines machining-specific features as "material to be removed by 2-1/2 axis production milling based upon "shop floor" terminology and that are independent of tool path creation". They are Shape features or groupings of geometric and topological entities from a component that correspond to primitive shapes produced by given manufacturing operations and tools. . Machining-specific features define the material to be removed, based upon machining geometric shapes such as pocket, slot, profile etc and are independent of tool path creation. A Feature defines not only the volume of material to be removed, it also contains manufacturing information, such as where the tool can safely and efficiently enter and exit the material. In Expert machinist the feature boundaries are defined in terms of "Soft walls" and "hard walls" .The "soft walls" and "hard walls" indicate stock (work-piece) and reference part surfaces respectively (Fig. 4). The hard walls are the walls that are part of the machining model and the soft walls are the ones that belong to the work piece.

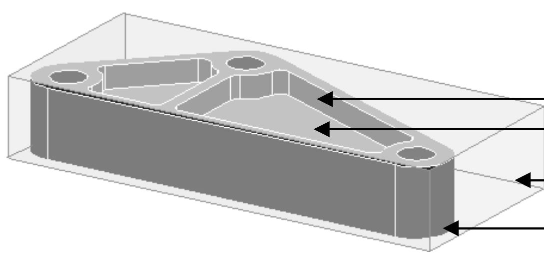


Fig. 4. The assembly of bracket part and its work piece with geometric characteristics of Pocket feature.

There are nineteen manufacturing (milling) features defined in Expert machinist and out of this table 1.0 gives the definition of 3 machining features.

Feature	Description
Face (F1)	The Face feature establishes the top of the part at the Z-level of its floor, with respect to the coordinate system active. A Face consists of a hard floor and a single, closed loop of soft walls. The soft walls that form the boundary are the outermost set of soft walls that form a closed loop.
Pocket (F2)	The Pocket feature is used for material removal where in respect to the coordinate system the removal takes place entirely within the periphery of the part. A Pocket consists of a hard floor and a single, closed loop of hard walls.

Slot(F3)	The Slot feature is used when it is desired to use a cutting tool diameter, equal to, or slightly smaller than the width of the feature. All forms have a hard floor. Form 1: consists of all hard walls. Form 2: consists of a single chain of hard walls and a single chain of soft walls forming closed loop.

Tab. 1. Some Expert Machinist milling Feature Definitions.

2.4 Pro/Toolkit

Pro/Toolkit can be described as a software tool through which details and information of any Pro/Engineer application being design, manufacturing or analysis, can be retrieved and output in a controlled and customised way depending on the needs of the user. Pro/TOOLKIT the C-language customisation toolkit for Pro/Engineer uses an Object-Oriented style and provides a large library of C functions that enables the external application to access the Pro/ENGINEER database and user interface, then integrating any resulting application into Pro/Engineer. The customised Pro/Toolkit library functions are used to retrieve information and data from the design model. A Pro/TOOLKIT object is a well-defined and self-contained C structure used to perform actions relevant to that object. Most objects are items in the Pro/ENGINEER database, such as features and others are more abstract or transient like ProSelection the information resulting from a select action. Fig. 5. (a) shows a list of objects such as ProAxis, ProCollection, ProEdge, etc and each object has its own set of functions and application purpose. In Fig. 5 (b) the functions associated with ProAxis object are listed.

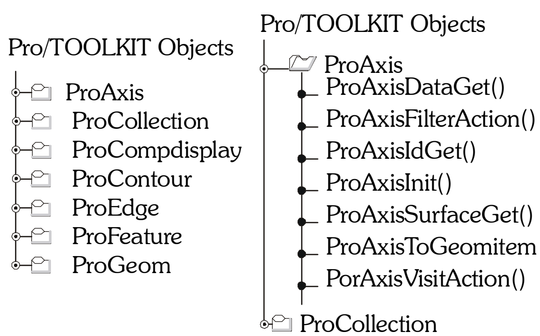


Fig. 5. Pro/TOOLKIT Objects and their functions.

Pro/TOOLKIT objects have a hierarchical relationship that reflects the Pro/ENGINEER database as shown in Fig. 6. By visiting geometry objects in a solid model the object handles of all the geometry objects can be acquired, either in the form of a ProGeomitem, or in the form of various specific opaque handles as ProEdge, or ProSurface.

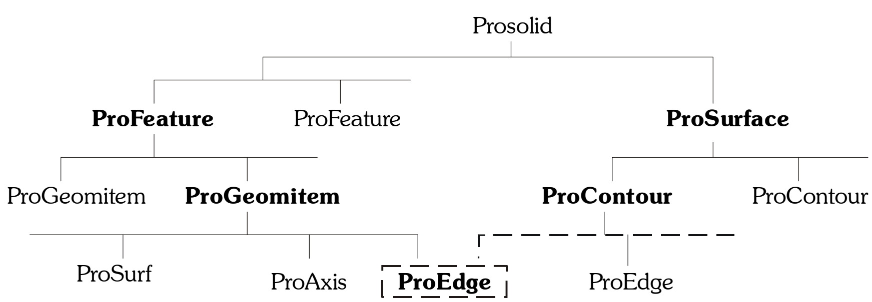
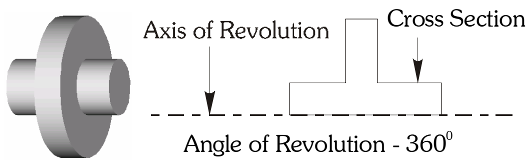
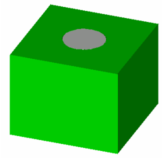
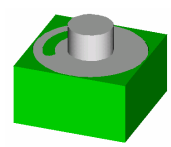
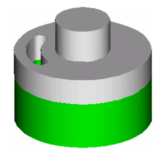
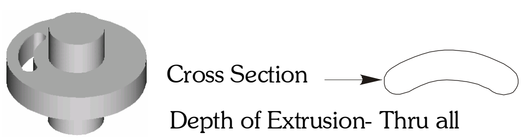
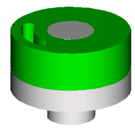
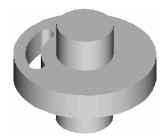


Fig. 6. Traversal of design features and their geometric entity objects in Pro/Toolkit.

2.5 Mapping

Once the design has been created using design features they must be mapped to manufacturing features to check its manufacturability and to carry out process planning. To transfer product model information from design oriented feature model to a manufacturing oriented feature model, the model has to be transformed from one viewpoint to another. In feature mapping, application specific features (manufacturing features) are recognized from another set of design features as shown in Table 2. It gives the steps in arriving at the artefact in both design mode and manufacturing mode. In design mode (a) shows the base feature creation by revolving the cross section through 360degrees. The second design feature (b) cut that's created by extruding the cross section through all. In manufacturing mode the part (light grey) and the raw stock (green) are assembled and features are added one by one starting from (a) face feature,(b) step feature, (c) thru pocket and (d) second face and (f) step features on the other side.

Design Features	Manufacturing (Milling) Features
 (a) Base Feature: Protrusion Revolved	 (a) Face Pocket  (b) Step  (c) Thru
 (b) Cut Feature: Extrusion	 (d) Face  (f) Step

Tab. 2. An illustration of design and manufacturing feature of the same model.

2.6 Requirement and Specifications of the Mapping Process

1. The mapping process should essentially do the following:
2. Understand the design intent of the model from the design feature structure.
3. For each feature or group of features, establish all the possible different set of sequences to be accomplished in order to manufacture the part.
4. Evaluate the order in which the sequences are to be accomplished. Indeed, the order may affect accessibility and available fixture surfaces.
5. Help in the choice of the manufacturing method to be applied by providing the designer with a quantitative and qualitative evaluation of the different available strategies.
6. Each Design Feature in the CAD model of the component should be realisable by manufacturing features supported by the intended machine and thus constructing a Manufacturing Feature Tree, which can realise the component described by the CAD model.
7. It should identify the direction of the normal for the manufacturing feature thus identifying the Tool Direction
8. It should identify and highlight the design features that cannot be manufactured with a specified setting-up in a machine.
9. It should be able to identify the number of set-ups that are needed to manufacture a component in a given machine.

3. METHODOLOGY

3.1 High Level Mapping

When analyzing a design for its machinability, all the manufacturing parameters needed for process planning need not to be considered. It needs a high level Feature Mapping only. Most manufacturing features require more than one operation in creating them. For instance Drill hole is a single manufacturing feature that might require three different operations, namely the access hole creation, drilling and finally threading. These detail analyses are only helpful in process planning stage and not for analysing the design of the component for manufacturability. What is important in manufacturability analysis is the type of manufacturing feature (hole), its orientation on this particular set up and its accessibility.

The number of setting-ups that are required and the machining features that can be manufactured in each setting-up varies according to the capability of the machine tool used. Thus the manufacturing feature, its face normal vectors,

accessibility and the machine coordinate system for the particular set up are the factors that need to be considered by the designer as opposed to all the details of cutter diameter, length of it, number of teeth, chamfer length, rough cut or finish cut and depth of cut as considered by the CAPP developer. Thus the requirements of the characteristics need to be considered can be generalised as (a) the type of feature (b) the orientation of the features and (c) its accessibility.
For mapping the generic and specific properties of both design features and manufacturing features have to be matched and if it satisfies certain conditions then mapping is carried out. Fig. 7 shows the generic properties of the hole feature creation starting from the type of hole, its diameter, placement plane, two references to place hole centre and its distance from the references and depth of the hole. Fig. 8(b) shows the manufacturing viewpoint of the hole feature and the important factors that affects it such as set up, feature accessibility and tool parameters. The created hole's geometric entities such as edges, faces with their properties such as shape, orientation together with traces of contours and edges shown in Fig. 9. These geometric details with feature topology are also used as the basis for mapping.

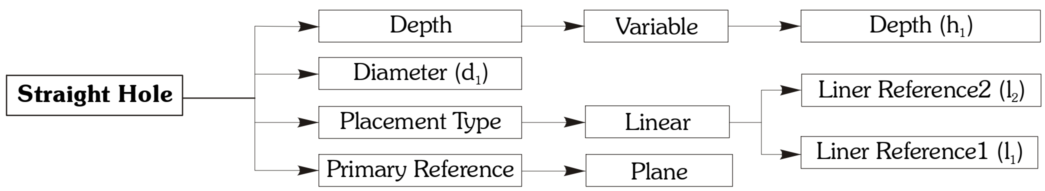


Fig. 7. Options used for straight hole creation in Pro/Engineer design mode.

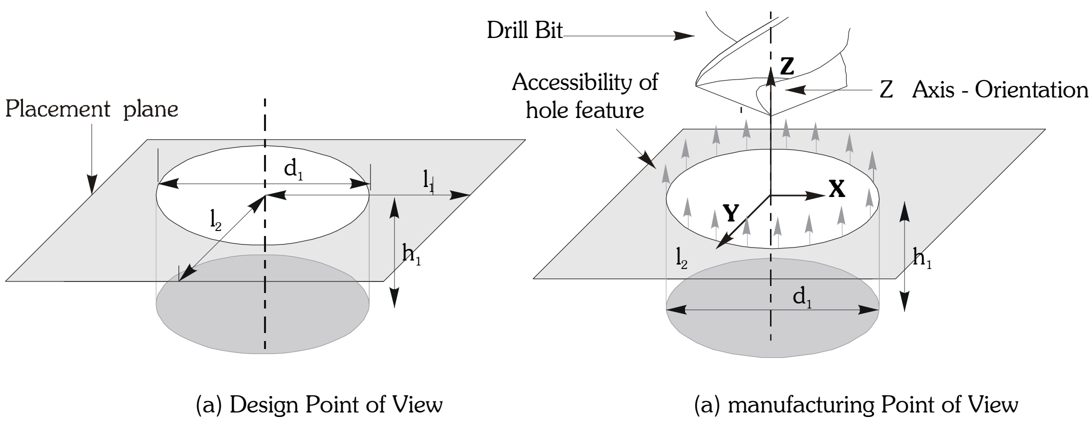


Fig. 8. Hole feature and its different Perspectives.

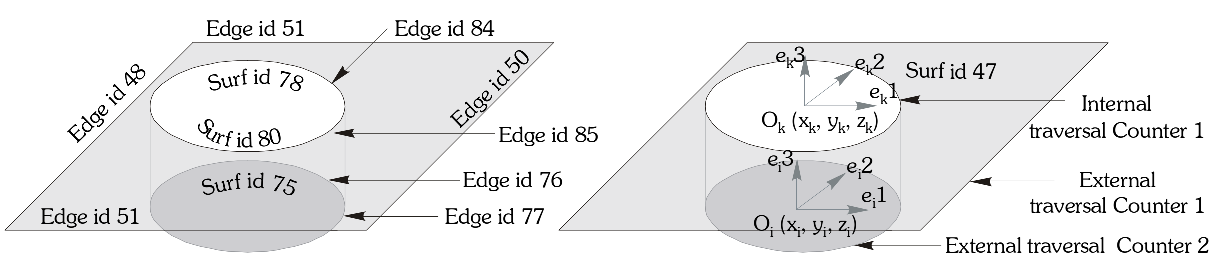


Fig. 9. The geometric entities, traces and their properties left by the Design Hole feature .

Each design feature created leaves feature hint or trace that is a characteristic trace left on the boundary of the part. These traces together with the feature properties are the input elements for algorithms in mapping the predefined manufacturing features. Based on each of the Pro/Engineer manufacturing feature's walls and floor condition definitions, the necessary and sufficient condition for their generic and specific geometric properties and topology are determined. These geometry and topology conditions form the basis for grammar in creating the set of predefined manufacturing feature templates. During mapping process these templates are matched against the design features and their properties, traces to arrive at the suitable manufacturing features. Once the design feature and its traces point to a particular manufacturing feature's existence it is checked for accessibility for particular machining setting using machining Coordinate.

4. IMPLEMENTATION

Pro/Toolkit provides the header files and functions written in C that enable the external application to access the Pro/Engineer database and retrieve and manipulate data through external applications written in Visual C++. Programs are written to access and manipulate the data and information about the model built in Pro/Engineer. Rules were established and algorithms written to interpret and associate data for deriving the manufacturing feature model from the design model.

The implementation is carried out in two stages. The first stage is identifying features that are machinable in a particular orientation and finding the features in order of machining sequence to ensure accessibility. The second stage is mapping the corresponding design features against the predefined set of manufacturing feature templates.

4.1 Stage 1 implementation

Flowchart given in Fig. 10 shows the sequence of activities of stage 1. These activities are explained in section followed by its implementation through Pro/toolkit functions and codes in C programming. The Pro/Toolkit Functions Used for Stage one is listed below for clarity.

1. Retrieving the Z vector of the Coordinate system that represents the machine coordinate

(e.g) Code: ProSelectionModelitemGet(sel[0], &csys_geom), ProGeomitemdataGet(&csys_geom, &gdata);

2. Scan through the surfaces of the feature model and retrieve the faces that have unit normal (e3) that are parallel and with same orientation as the z-axis vector

Code: status = ProUtilCollectSolidSurfaceByVector(solid, gdata->data.p_csys_data-> z_vector, USER_PARALLEL, &p_surface);

3. Arrange the surfaces in a descending order of distance from machine coordinate to represent the sequential machining order of the surface

```
Code: x = ProUtilPointPlaneDist( gdata->data.p_csys_data-> origin, p_gdata ->data.p_surface_data-> srf_shape.plane.origin,
p_gdata ->data.p_surface_data-> srf_shape.plane.e3);
status = ProArrayObjectAdd((ProArray*)&distance_array, PRO_VALUE_UNUSED, 1, &x);
l = index_of_min ( distance_array, surface_num);
```

4. Scanning individual surfaces to extract the design features that open up on them and map them against machining feature templates to identify the presence of manufacturing features in the model.

Code:

```
ProUtilCollectSurfaceContours( p_data->surface, &p_contours );
status = ProContourTraversalGet(contour, &traversal);
status = ProUtilCollectContourEdges (p_data1->surface, contour, &edges);
status = ProEdgeToGeomitem(p_data1->part, edges[i], &geomitem);
status = ProGeomitemFeatureGet(&geomitem, &feature);
status = ProFeatureTypeGet (&feature, &featype);
status = ProUtilCollectFeatureGeomitems( &feature1, PRO_SURFACE , &p_geomitems );
```

4.2 Stage 2 implementation

In stage two the individual manufacturing features have to be identified for chosen set up. This involves finding the presence of the features from their traces and checking their properties against manufacturing feature template to identify the particular instance of manufacturing feature. Fig. 11 shows the Flow chart for face feature identification. Similarly algorithms were written for other manufacturing feature to facilitate their recognition from design model.

5. CASE STUDY

In order to help the designer at the design mode itself the proposed system is integrated to the design mode of Pro/Engineer by means of additional menu bar that's docked on the main menu bar of Pro/Engineer. Each time a designer has to check for the conformity of his design feature with any one of the manufacturing features he has to click on the "manufacturing feature_List" that is docked on the top tool bar (Fig. 12). The individual machining feature analysis options are given as a drop down menu from the "manufacturing feature_List" menu. Once the required manufacturing feature is option is selected the prompt to "pick" the machine coordinate representing a particular setting is required. This will identify all machining features that are on the design model and machinable in that setting. Fig.13 illustrates the identified machine features for selected machine coordinates.

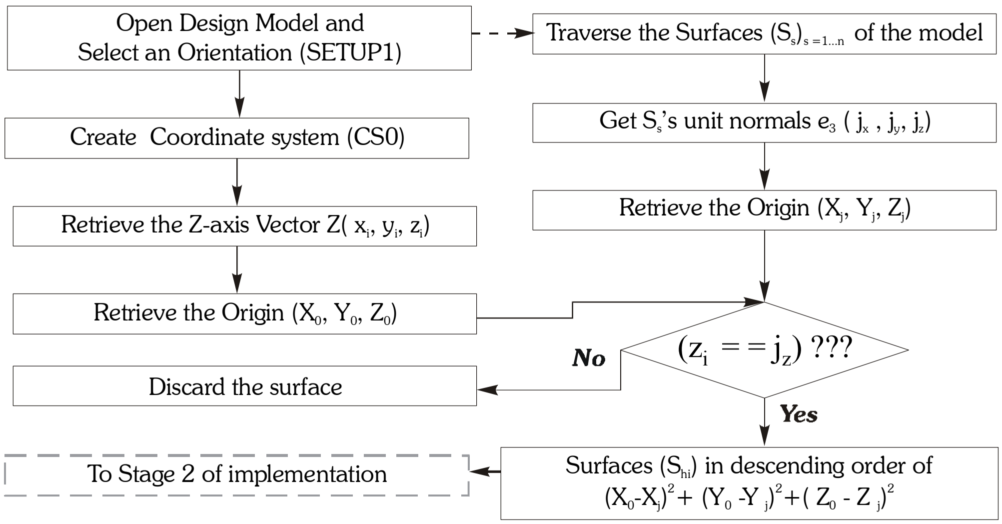


Fig. 10. Flow chart of stage 1 implementation.

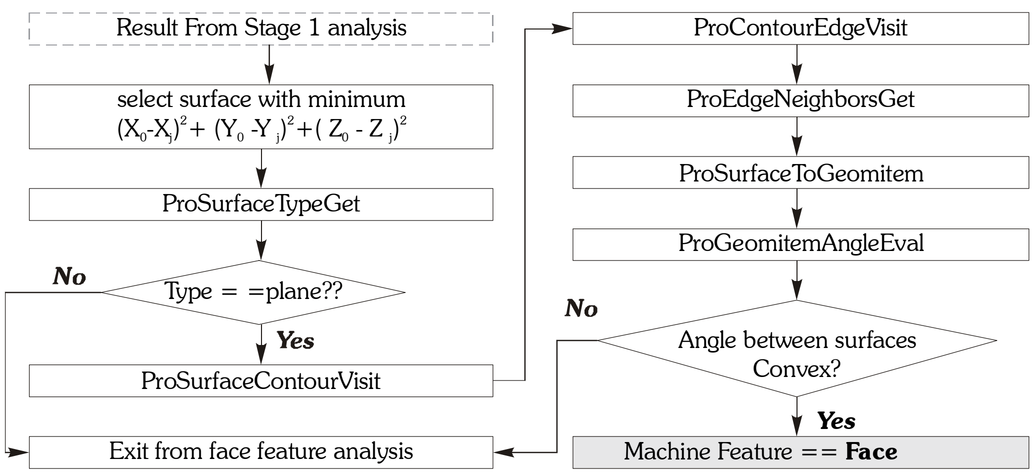


Fig. 11. Flow chart of for face feature mapping.

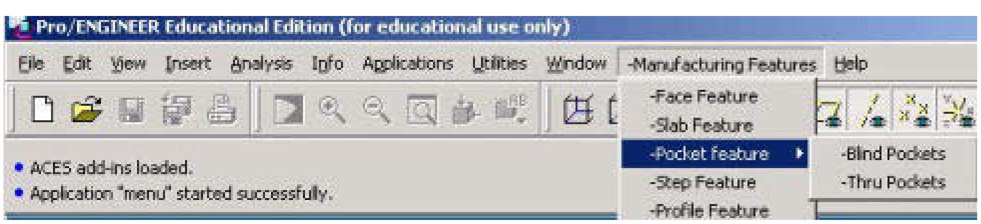
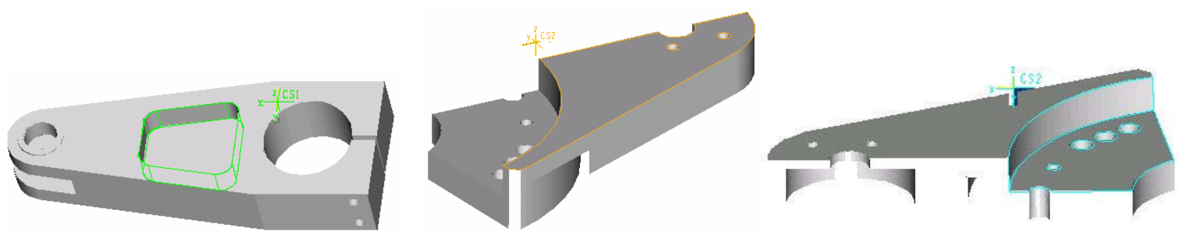


Fig. 12. Sub menus of manufacturing feature menu giving the set of manufacturing features in of expert machinist.



(a) Blind Pocket (b) Face (b) Slab

Fig. 13. Manufacturing features recognized for the selected machine feature type and machine coordinate.

6. REFERENCES

[1] Aritstides A G Requicha, Representations for Rigid Solids: Theory, Methods and Systems, Computing Surveys, Vol. 12, No.4, Dec 1980.

[2] Martti Mäntylä , An introduction to solid modelling, Computer Science Press, 1988.

[3] Jami J. Shah and Martti Mäntylä Parametric and feature-based CAD/CAM: concepts, techniques, and applications, John Wiley & Sons Inc, 1995.

[4] Martti Mäntylä, Dana Nau and Jami Shah, Challenges in feature-based manufacturing research, Communications of the ACM, Volume 39, Issue 2, 1996, pp 77–85.

[5] Chang-Xue Feng, Chun-Che Huang, Andrew Kusiak and Pei_gen Li, REPRESENTATION OF FUNCTIONS AND FEATURES IN DETAIL DESIGN, Computer-Aided Design, Vol. 28, No.12, 1996, pp 961-971.

[6] S. Joshi and T C Chang, Graph based heuristics for recognition of machined features from 3-D Solid Model, Computer Aided Design, Vol. 20, No. 2, 1988, pp 58-66.

[7] P Gavankar and M R Henderson, Graph based extraction of protrusion and depression from boundary representations, Computer Aided Design, Vol. 22, No 7, 1990, pp 442 – 450.

[8] Leila De Floriani, A graph based approach to object feature recognition, Source Annual Symposium on Computational Geometry archive, Proceedings of the third annual symposium on Computational geometry table of contents, Waterloo, Ontario, Canada, 1987, pp 100 109.

[9] E Bruzzone and L De Floriani, Extracting adjacency relationships from a modular boundary model, Computer Aided Design, Vol. 23, Issue 5, 1995, pp 344-355.

[10] M Marefat, R L Kashyap Geometric Reasoning for Recognition of Three-Dimensional Object features, IEE transactions on Pattern Analysis and Machine Intelligence, Vol. 12, Nos. 10, 1990, pp 949- 965.

[11] Sanjeev. N. Trika, R. L. Kashyap, Geometric Reasoning for Extraction of Manufacturing Features in Iso-Oriented Polyhedrons, Source IEEE Transactions on Pattern Analysis and Machine Intelligence archive, Vol.16, Issue 11, 1994, pp 1087 – 1100.

[12] Bernard M Chazelle, Convex Decompositions of Ployhedra, STOC –Milwaukee, 1981, pp70-79.

[13] K Tang and T Woo, Algorithmic aspects of alternating sum of volumes. Part 1: data structure and difference operation, Computer Aided Design, Vol. 23, Nos.5, 1995, pp 357-366,.

[14] Y S Kim, Recognition of Form features using Convex Decomposition, Computer Aided design, Vol. 24, Nos. 9, 1992.

[17] Hiroshi Sakurai, Volume Decomposition and Feature Recognition: Part1- Polyhedral Objects, Computer-Aided Design, Vol.27, Nos.11, 1995, pp 833-843.

[18] Jan H. Vandenbrande, Automatic Recognition of Machinable Features in Solid Models, PhD Thesis, Computer Science Department and Institute for Robotics and Intelligent Systems, University of Southern California, 1990.

[21] Williams C. Regli, Geometric Algorithms for Recognition of Features from Solid Models, PhD Thesis, University of Maryland, 1995.

[22] Ryszard Jakubowski, Extraction of shape Features for Syntactic Recognition of Mechanical parts, IEEE transactions on Systems, Man and Cybernetics, Vol 15, No 5, 1985, pp 642- 651.

[23] Satyandra K Gupta, Thomas R Kramer, Dana S Nau, William C Regli and Guangming Zhang, Building MRSEV models for CAM applications, Advances in Engineering Software 20(2/3), 1994, pp121-139.

[24] Thomas R. Kramer and Fedrick M. Proctor, Feature based control of machining centre, NISTIR5926, 1996.