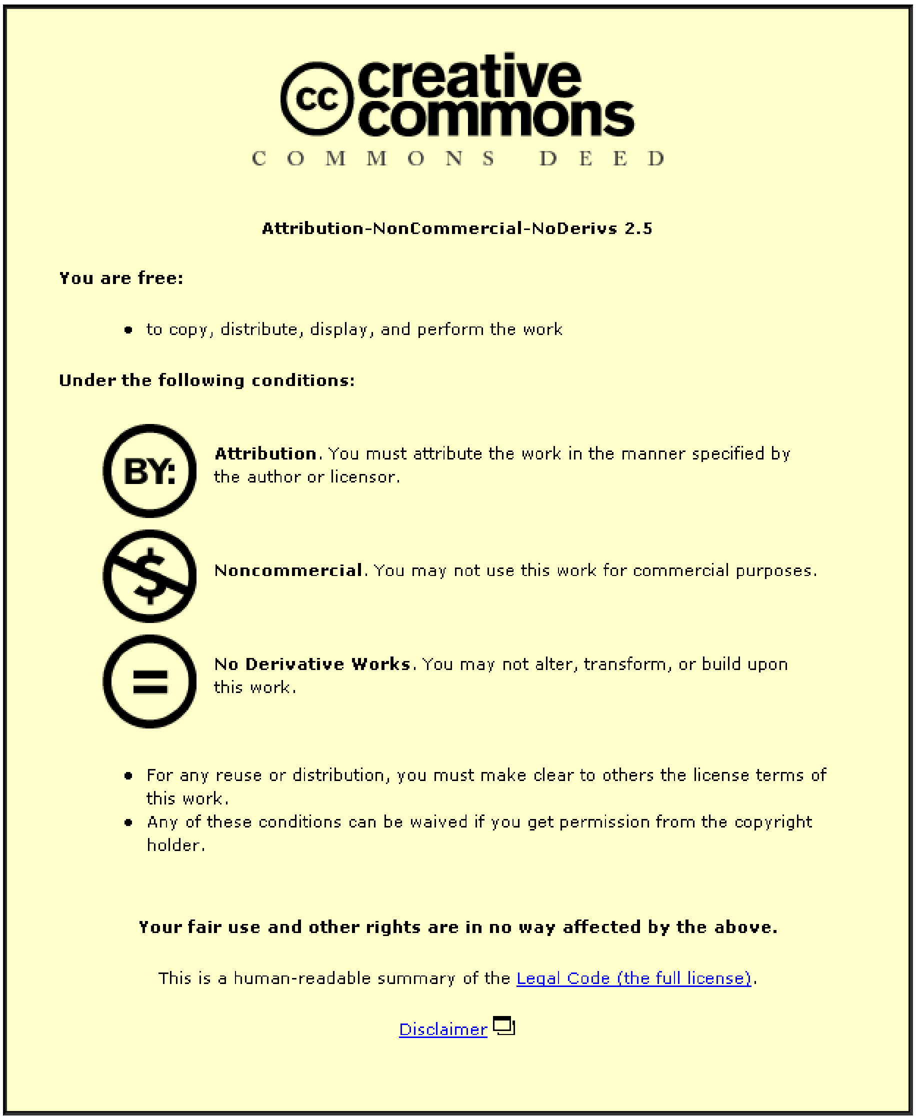

International Journal of Computer Integrated Manufacturing
Vol. 6, Nos. 1 and 2, pp2-12, 1993

Feature technology: an overview

K. CASE and J. GAO

Abstract. The proper integration of the activities of Computer Aided Design and Computer Aided Manufacture is an objective that has become more urgent within the wider context of a total Computer Integrated Manufacturing environment. In seeking this integration it is recognised that the diversity of activities and consequent needs for data can best be served by a single representation for design, design analysis and manufacturing planning, and that a strong candidate for this descriptive role is a Feature Representation. This paper briefly overviews the primary methods of the use of features through Feature Recognition and Design by Features, particularly in the process planning application area.

1. Introduction

The use of features is seen by many researchers as the key to the genuine integration of many aspects of design and the planning of manufacture. On the design side this could relate to the fulfilment of functional requirements, the building of a geometric model or as preparation for design analysis activities such as finite element analysis. On the planning side activities such as process planning, assembly planning, inspection planning, manufacturing operations planning and part programming for numerically controlled machines could potentially be based upon a feature representation of the component. In a computer-controlled environment, the integration of these aspects of design and manufacturing planning are considered to be beneficial in its own right, but there are of course even wider implications for the integration of the CAD and CAM technologies into company-wide activities as part of a total Computer Integrated Manufacturing (CIM) environment.

Research into the use of features in this way is now a mature body of work, but there are still considerable divisions as to the particular features approach to be adopted and their use in particular fields of application.

Attempts to define the precise nature of features are fraught with difficulty because of the wide interpretation placed upon the term by different researchers. Hence it is perhaps useful to first identify the need to go beyond the capabilities of a purely geometric representation.

One of the attraction's of geometric modelling within CAD is that a well-constructed modeller is capable of representing all the geometric aspects of a component within the chosen domain. It is thus theoretically possible to extract any of this information from the CAD model and use it for example in manufacturing planning. Given an adequate domain, such as that possessed by solid modellers (Requicha, 1980), there would then seem to be no need to seek alternative representations, only a need for a range of different interpretations. However, a purely geometric description of a component is rarely adequate in itself to allow the planning of manufacture. For example, a total knowledge of the geometric condition of the external surfaces of a component is a necessary but inadequate set of data to plan the machining of it. In addition it is essential to have knowledge of relationships between the surfaces, particularly how they are grouped to represent aspects of the component which will be generated by a particular machining operation, how such groups relate to each other on the component and how manufacturing information is associated with the geometry. A specific example of the above would be a cylindrical surface associated with a plane surface to represent a blind round hole, a pattern of such holes (on a pitch diameter for example), and the tolerance and surface finish requirements. The blind hole could be described as a *feature*, the pattern of holes as a *compound feature* and the manufacturing information as *feature attributes*.

One difficulty with the above rather loose description of a feature is that it is particularly associated with the needs of machining and is less suitable for other application areas. ie the designer may wish to express this as a location, the assembly planner as an assembly feature, the inspection planner as a recess, and so on. Here lies one of the major divisions between researchers in the field, where some are happy to allow each application to use a different features approach (despite the needs of integration), some permit this division but seek to heal it with mapping between the different applications and others seek a unifying feature representation to suit the needs of all applications.

This last approach of a unifying *Feature Representation* within a computer modelling environment would appear to offer the most for the future. If features are to be the bridge between design and manufacturing planning then a representation is required that meets the needs of both sides without compromising the objectives of either. Formal methodologies for feature representation are very much at the research stage and as yet no particular approach has fulfilled all requirements. A review of some of the approaches is given in this paper.

Returning to the idea of a geometric model that has complete knowledge of the appropriate physical aspects of the component, the second major division in the research work can be found. If a complete geometric description is available, then it is clearly possible to use computer methods to interrogate this information and transform it into any desired form. Thus collections of surfaces could be recognised as features. Much work has been done in this *Feature Recognition* field, but it is perhaps fair to say that very much more remains to be done. The problems arise from the complexity of three-dimensional geometry and the diverse ways in which it must be interpreted. An alternative approach is to collect the surfaces together at the time of creation (design) rather than try to synthesise it on completion of the geometric specification. This *Design by Features* approach attempts to capture the designer’s intent rather than ’second guess’ it, and is generally achieved by constraining the designer to construct his component design from a limited set of pre-defined features.

Both of the methods are subject to legitimate criticism. Feature recognition methods usually have very limited domains because of our as yet inadequate ability in the application of geometric reasoning techniques. Once this problem has been overcome however the approach does allow the designer the freedom to use a geometric modelling system as he sees fit to achieve the functional requirements of a design. A more fundamental criticism is the wanton abandonment of design intent by the use of the final state of the geometric specification as the only design output, whereas in reality the designer will have explicitly or implicitly generated much other useful information.

Design by Feature is potentially helpful in overcoming this latter problem but also typically suffers from a limited domain, which often leads to it being criticised for over-constraining the designer. However a counter argument would support the constraint of the designer as a method of implementing company standards and design for manufacture objectives.

A number of excellent review papers have been published in the last five years such as Alting and Zhang’s (1989) review of Computer Aided Process Planning systems. Brimson and Downey (1986), Pratt (1988), and Shah and Rogers (1988) have discussed feature technology and its role in the integration of CAD and CAM. Jared (1989a), Joshi and Chang (1990) and Woodwark (1988) have discussed the different feature recognition approaches. The authors (Gao and Case, 1991) have reviewed feature representation methodologies and the following sections are a summary of this work.

2. Feature terminology

Features originate in the reasoning processes used in various design, analysis and manufacturing activities (Cunningham and Dixon 1988) and are frequently strongly associated with particular application domains. Hence there are many different definitions for features. A broad definition in the engineering domain is given by Pratt and Wilson (1985) as: ”A feature is a region of interest on the surface of a part”. Some definitions are related to the representation and recognition methodology such as Henderson’s: ”Features are defined as geometric and topological patterns of interest in a part model and which represent high level entities useful in part analysis” (Henderson et al, 1990). Herbert et al (1990) define a feature as a group of geometric entities with some meaning for the particular activity to be performed with them. Choi et al (1984) relate features to the manufacturing methods and define a feature as a portion of the workpiece generated by a certain mode of metal cutting.

In the design process, a feature is considered by the designer as a ’design feature’, in terms of its geometry, specifications and details to fulfil certain functional requirements, and thus is sometimes called a ’functional feature’. Examples of such features are fixing holes, keyways and cooling slots. However, features may be viewed differently by process planners or NC programmers as ’manufacturing features’. For example, a fixing hole may be considered as a drilled or bored hole; a cooling slot may be considered as a general slot machined by a slot cutter, etc. For another example, see figure 1. Even within the manufacturing domain, different views may be taken of features. For instance, in metal cutting, a feature may be considered as the volume of material to be removed or a ’volumetric feature’ which is of a negative nature, whilst for injection moulding or casting, a feature is usually considered as the volume to be added and therefore is of a positive nature. When the geometry of a feature is being considered, a feature is usually called a ’form feature’, a ’shape feature’ or a ’geometric feature’ (Jared, 1986).

3. Feature taxonomy schemes

In practice, features are usually divided into different classes so as to help the designer to access the feature data and the manufacturing engineer to generate process plans for a group of features which have some common geometric, topological or other properties. Such classes can be further divided into sub-classes such that classes and sub-classes form a hierarchy. This classification structure is known as a feature taxonomy. Since the taxonomy of features is of a hierarchical nature, the attributes of a class can be inherited by its sub-classes. The way of classifying features is highly dependent on the feature representation methodologies and the strategies for the eventual use of the feature data (eg as input into a process planning system).

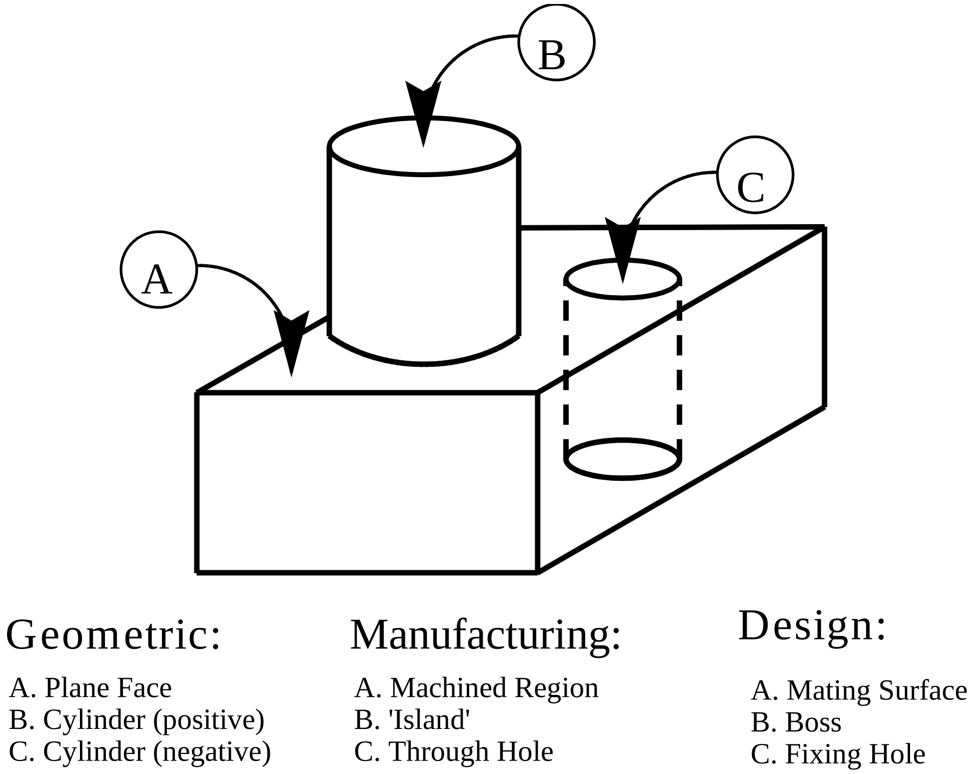

Figure 1. Different interpretations of the same geometry

Butterfield et al (1985) classified form features into three main categories: sheet features, non-rotational (prismatic) features and rotational features. Sheet features were further classified as flat or formed (flat patterns were further classified as depressions, edges, etc. and formed features were classified as localized and non-localized); Non-rotational features were classified as depressions, protrusions and surfaces (depressions can be internal, external, through and non-through, etc.); Rotational features were classified as concentric and non-concentric. Since this scheme was intended to be the standard for all the application programs carried out in the CAM-I (Computer Aided Manufacturing International) project, it is broad and general.

Pratt and Wilson (1985) divided feature representations into two types, i.e., explicit (where all the geometric details of a feature are fully defined) and implicit (where sufficient information is supplied to define the feature but the full geometric details have to be calculated when required). Simple explicit features were classified into four main classes: through holes, protrusions, depressions and areas, with possible sub-classification in terms of their cross-sectional shapes as rotational and prismatic. Hummel and Brooks (1986) applied a similar taxonomy to their expert process planning system - XCUT, using an object-oriented programming language for the implementation.

Gindy (1989) treated form features as volumes enveloped by entry/exit and depth boundaries. The feature classification is based on the External Access Directions (EADs, see figure 2) from which the feature volume could be machined by cutting tools. Form features are first divided into three categories, ie. protrusions, depressions and surfaces (Figure 3). Feature geometry is described by defining the EADs, the boundary type (open, closed) and the exit boundary status (through or not through). The result of grouping features according to these characteristics is a list of form features classes or primary features, such as bosses, pockets, holes, slots, notches, and real and imaginary surfaces. This scheme is closely linked to the process planning requirements and is sufficient to classify the features used in this domain.

Gandhi and Mycklebust (1989) used a parametric approach to the definition of features. The taxonomy of features is based on the topology of feature primitives, ie. features having the same topology are grouped together so that they could be defined using the same number of parameters. Thus for example, topology group A is defined by a radius and a length and can include features such as ’bar’, ’cylinder’, ’disk’ and ’cylindrical plate’. Features can additionally be classified according to ’form’ such as ’rotundity’, ’angularity’, ’curvature’,

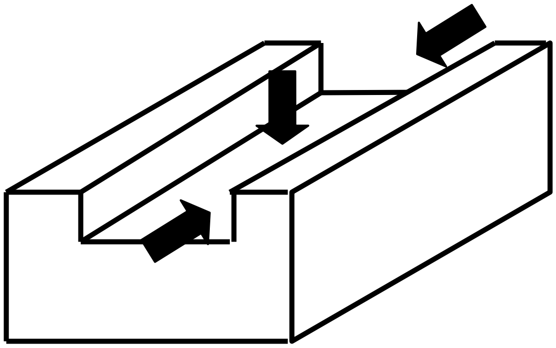

Figure 2. External Access Directions (EAD's) for a through slot.

'straightness' and 'circularity'. A descriptive language including for example 'shape transforming verbs' combined with the classification methods identify this parametric approach with design rather than manufacturing activities.

Taxonomy schemes have the primary purpose of structuring information in a way that makes subsequent processing easier. Thus Butterfield's taxonomy is general and could be used for different applications such as process planning and NC programming while Gindy's scheme is specifically aimed at the needs of process planning in a machining environment. Without a rigourous taxonomy it is very difficult to produce analytic and predictable algorithms for the complex task of manufacturing planning. Hence loosely defined feature classifications frequently lead to the necessity to use unsatisfactory heuristics or 'rules of thumb'. A secondary objective of feature taxonomies and the associated feature representations is to provide a framework for the parametric generation of geometry at the design stage. Hence both Pratt and Wilson's and Gandhi's schemes are suitable for representing features in a solid modelling environment.

4. Feature representation methodologies

In the design and manufacturing environment a part can be described in a number of ways, typical methods being engineering drawings (two dimensional), physical models (clay models, wooden prototypes and templates), GT (Group Technology) codes, symbolic representations and the modern computer-based geometric representations: wireframe, surface and solid models (Chang, 1989). Some of these methods are briefly described below.

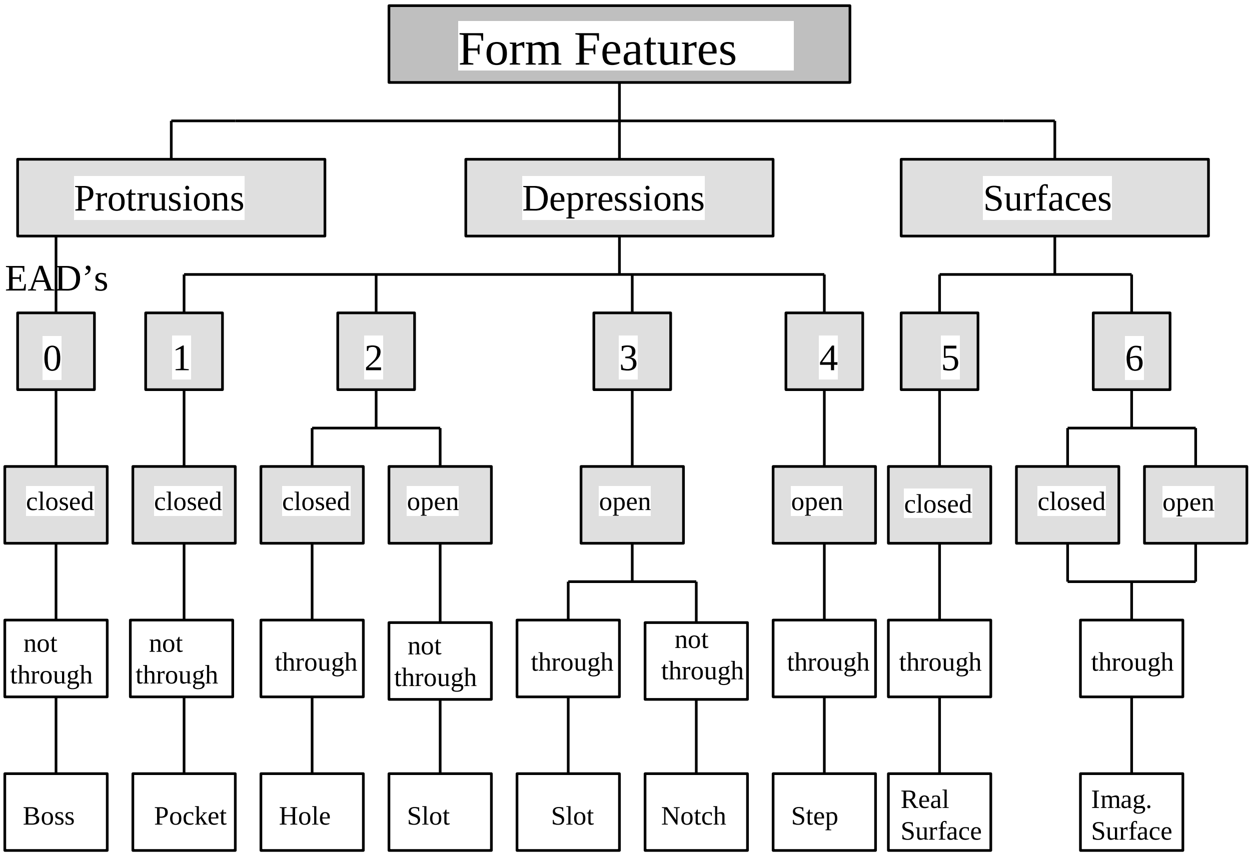

Figure 3. Gindy's Feature Taxonomy.

4.1 GT Coding

GT part coding is a technique in which similarities are used to group parts into a family either because of geometric shape and size or process requirements (Henderson, 1986). The method is mainly used in variant process planning for standard process plan retrieval. Since a code has to represent a large number of similar parts, it does not contain sufficient detail to uniquely define a particular part.

4.2 2D Representation

Two dimensional representations do not provide a complete description of real parts, and human assistance is needed to reason about the complete geometry and details. Despite this, some systems still extract feature information from 2D models (Wang, 1987, Wang and Wysk, 1987, Mortensen and Belnap, 1989, Lui et al, 1991) or apply 2D feature representations for rotational parts (Joseph et al, 1990, Varvarkis, 1991).

4.3 3D Wireframe

3D Wireframe models are also incomplete representations (see figure 4), but they are sometimes used for quick display and verification of the geometry of feature models (Luby et al, 1986, Shah and Rogers, 1990).

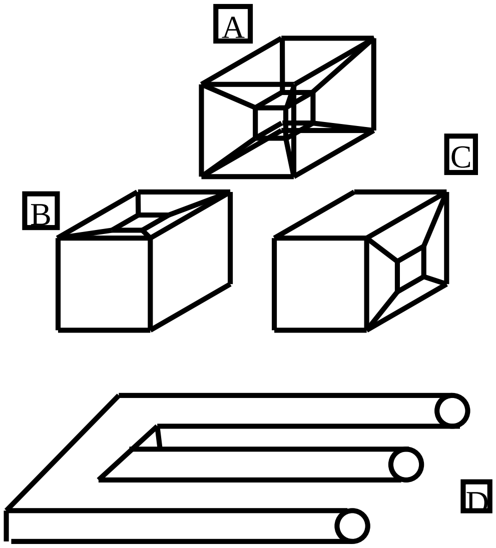

Figure 4. Wireframe models can be ambiguous (A could be interpreted as B or C), or simply impossible (e.g. the ‘devils tuning fork’ shown as D).

4.4 Surface Models

Surface models do not represent the volumetric properties of geometric objects (which are essential for process planning and NC machining where features may be defined as removal volumes). However, surface modellers can represent and manipulate objects composed of free form surfaces. Gandhi and Myklebust (1989) represent feature geometry using a combination of solid models (boundary representation) and surface models (non-uniform B-splines) to extend the complexity of the shapes that can be modelled.

4.5 Symbolic Representations

Symbolic representations of parts describe objects in descriptive languages. For example, Hart (1986) used a ’familiar engineering language’ which was capable of describing rotational components as a sequence of manufacturing operations (feature generations) to be carried out on some stock material. This ’outline process plan’ provided a starting point for detailed process planning and was also used to drive a CSG modeller to obtain geometric representations. Most symbolic representations are feature-based and therefore they are widely used in process planning, especially in frame-based knowledge-based expert systems (Hummel and Brooks, 1986, Chung et al, 1988, Shah et al, 1988a).

Feature objects are usually stored as frames in knowledge based systems. A frame is a data structure which represents an entity type (Frost, 1986), and consist of a collection of named ”slots” each of which can be ”filled” by values or by pointers to other frames. Frames may be linked to other frames in various ways such as by filling one frame by another, by a pointer to another frame, or by linking in taxonomical structures.

The advantage of using the above approach is that the complete feature model representation can be achieved, ie. feature geometry, topology, dimensions and tolerances and all the other attributes can be represented as frames in a descriptive way; feature relationships such as geometric constraints, parent-child relationships, and feature taxonomy can be represented through the slots of the frames or the links between the frames. However, since it is not easy to use the symbolic representation scheme for CAD purposes, the process planning systems using this approach should be integrated with CAD systems, to avoid the need for manual input of feature information using the grammars of the object-oriented programming language.

4.6 Solid Models

Solid modelling defines the geometry and topology of parts completely, and is therefore used in many current

CAD systems. A geometric model with an internal representation that emphasizes the topological structure, with data pointers linking together an object’s faces, edges and vertices, is a graph-based model. The most widely used is a Boundary Representation (BRep). Alternatively, if the internal representation is by the Boolean combination of two or more simple objects (primitives), then the representation is considered to be a Boolean model. This is a procedural model (also called an unevaluated model), since only the way to construct the model is known, and the geometric data and topological characteristics of the model need to be determined by further evaluation. One of the most widely used Boolean modelling methods is Constructive Solid Geometry (CSG). CSG models contain higher level entities (primitives) which have some advantages for representing features.

4.6.1 BRep-based Approach

In the BRep context, features are defined as sets of related faces or ’facesets’ of a part (Jared, 1989b, Chuang and Henderson, 1990, Henderson et al, 1990). These features are also called ’surface features’. BRep models are graph-based and all the geometric and topological information is explicitly represented in the face-edge-vertex graph, and hence BRep models are often called evaluated models. The BRep approach has been favoured by a number of researchers (Turner and Anderson, 1988, Pratt, 1988) because sufficient information is available and because of the graph-based representations (many feature recognition systems are based on the graphs). BRep models can also be associated with attributes (e.g. surface finish, material) and dimensions and tolerances where face, edge and vertex information is essential (Requicha, 1990, Faux, 1986). The disadvantage of the BRep approach is that it has no meaningful link to the feature primitives and volumetric features. Feature operations such as deleting features (Shah et al, 1988b) are difficult to perform.

4.6.2 CSG-based Approach

CSG based feature representations define features as volumetric primitives which construct a part through Boolean operations (Nnaji and Lui, 1990, Case and Acar, 1989, Requicha, 1990, Shah and Rogers, 1989). The reasons for using the CSG representation are that it is concise, simple and powerful, is easy to edit and manipulate primitives, provides a meaningful link between CSG and feature primitives and the binary tree can be used for feature model construction. The main problem with the CSG models for feature extraction is their non-uniqueness and the lack of an explicit representation of lower-level entities such as lines, points and surfaces. However, these problems can be overcome by evaluating the CSG model to derive a boundary representation.

4.6.3 Hybrid CSG/BRep based Approach

Since both CSG and BRep representations have their advantages and disadvantages, the hybrid approach which exploits the advantages of both CSG and BRep has been considered. Nnaji and Lui (1990) have developed a process planning system which extracts the CSG-based information (with BRep information derived from the CSG model). The CSG tree and the BRep information are then reconstructed into a different CSG tree which is represented in a hybrid format to represent features. Roy and Lui (1988) proposed a hybrid CSG/BRep approach to representing features, especially the dimensions and tolerances. A hierarchical structure of features (as primitives of the CSG-tree) provides a multi-level representation of the object component relations (CSG) and maintains the boundary representation at each level of detail. Gossard et al (1988) presented a method for explicitly representing dimensions, tolerances and geometric features in solid models. The method combines CSG and BRep representations in a graph structure called an object graph.

4.8 Summary of Representation Schemes

From the above discussion, it can be summarized that 2D and 3D wireframe models are not complete representations of a part’s geometric and topological properties and can only be used for some specific purposes such as partial representations of rotational parts and displaying part geometry in a feature-based system. GT coding techniques are suitable for variant process planning systems and the general concepts are useful elsewhere, but the codes are not explicit enough for a complete part representation. Solid models are the most complete representations of part geometry and topology and are already widely used in CAD/CAM systems.

It can be concluded that the hybrid CSG/BRep scheme is the best approach to representing features in design systems, since it combines both the advantages of the CSG models (higher level primitives such as removal volumes and easy performance of feature operations such as add, delete and modify) and the advantages of the BRep models (lower level entities such as faces, edges and vertices for attaching dimensions, tolerances and other attributes). Successful process planning systems (usually knowledge-based) will often represent features using symbolic and object-oriented approaches which permit the representation of the geometry, topology, the relationships between features and the taxonomy of manufacturing features completely and offers geometric reasoning facilities. However, such a planning system needs to be integrated with a CAD system to avoid the need to input feature data manually.

5. Feature data models

The representation of a part in terms of features is known as the feature model of the part, and the associated database is known as the feature data model. Since the definition of features is frequently application-dependent, there are often different feature models for different applications, such as design feature models, manufacturing feature models and form (geometric) feature models. Two main approaches to creating such feature models are discussed here, Feature Recognition and Design by Features.

5.1 Feature Recognition

Use of the feature recognition approach for building feature models (Gavankar and Henderson, 1990, Chuang and Henderson, 1990, Brooks et al, 1987, Choi et al, 1984) allows the design of parts using conventional CAD systems such as two dimensional drafting systems, wireframe modellers and solid modellers. Form features are then extracted from the geometric models using a recognizer and are stored in a separate database which forms the feature model.

Jared (1991) has categorised recognisers according to various characteristics. Those that work on boundary representations (the majority), can either be edge or face based depending on the characteristics of the underlying solid modeller. The feature recognition process itself can be based on heuristics or loose ’rules of thumb’ or alternatively the process might be rigid and formalised, perhaps within the framework of a feature taxonomy. The method of implementation can vary from ’hard-coding’ to syntax directed. Syntax directed methods allow separation of the geometric rules from the algorithmic methods used to evaluate them in a way which is analogous to the separation of the knowledge and inference methods in knowledge-based applications.

The recognizer itself will be highly influenced by the underlying solid modelling technology with topological adjacency methods being favoured for BReps and pattern-matching techniques for CSG.

There are two main ways to achieve pattern matching, ie. syntactic pattern recognition and rule based template matching. In syntactic pattern recognition (Joshi and Chang, 1990, Chang, 1989), a picture is represented by some semantic primitives, which are written in a picture language. A set of grammars consisting of some re-write rules define a particular pattern. A parser is then used to apply the grammar to the picture. If the syntax of the picture matches the grammar, then the picture can be classified as belonging to the particular pattern class. In template matching (Mortensen and Belnap, 1989), parts are described in a symbolic form, and the rules in a pre-defined template look for certain patterns of elements and relationships until some set of elements can be matched and identified as a feature. A template is a Prolog predicate which consists of relationships satisfying a particular pattern to be matched. Each template usually corresponds to a feature (but could, for efficiency reasons, represent a composition of several features). The rule based approach is suitable for knowledge based expert process planning systems (Herbert et al, 1990, Mortensen and Belnap, 1989, Wang and Wysk, 1988 and Joseph et al, 1990).

Since BRep models contain sufficient information about faces, edges and vertices of objects, most systems extract features from the face-edge-vertex graphs of BRep models (or their sub-graphs). Choi et al (1984), van Houten et al (1989) and Jared (1989b) extract features from the complete graph of a BRep model; Henderson et al (1990) and Lee and Lee (1989) extract features from the face-edge graph (the sub-graph of the face-edge-vertex graph); Chuang and Henderson (1990) extract features from the vertex-edge graph; and Joshi and Chang (1988) introduced an approach of extracting features from an Attributed Adjacency Graph (AAG) built on the underlying BRep models. The main advantage of a graph-based approach to feature recognition is that only topological information of a boundary model is required for some types of extractions.

Extracting features from CSG models is not as easy as from BRep models, due to the non-uniqueness of CSG binary trees (figure 5). A part model may be represented by an arbitrary CSG tree (depending on different designers), requiring an unlimited number of shape grammars or templates to match the trees. To solve this problem, Lee and Fu (1987), Herbert et al (1990) and Hinde et al (1990) re-construct the arbitrary CSG tree to form a unique and computer understandable tree, then the nodes (primitives) of the re-constructed tree can be identified by a recognizer. Li and Yu (1990) and Perng et al (1990) converted the CSG tree into a DSG (Destructive Solid Modelling) tree, then features are recognized from the DSG model.

The feature recognition approach requires that each feature (or feature class) has a pre-defined pattern primitive or rule-based template. This limits the number of features (or feature classes) that can be recognized. Also, before the recognition process can begin, the part representations in the CAD systems must be translated into a description (in string form or symbolic form) based on the pattern primitives. For 3D models and complex components, the recognition process can be very complex and involve considerable computing time.

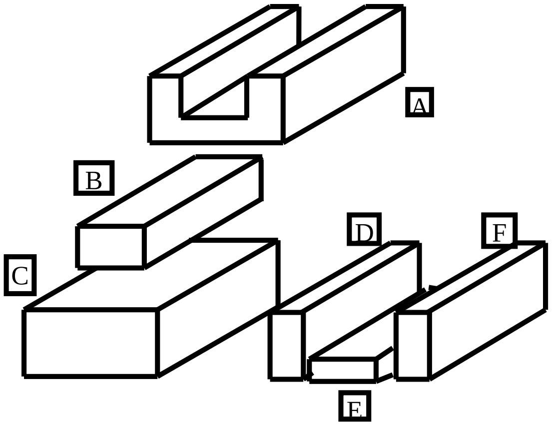

Figure 5. CSG Representations can be non-unique such that object A could be generated by a difference between objects B and C, or the union of objects D, E and F.

5.2 Design by Features

In the design by features approach (Dixon, 1988, Shah and Rogers, 1990, Chung et al 1988, Chang 1989, Case and Acar 1989, etc.), the designer is provided with a features library, similar to the primitives of a CSG system, which can be used with a set of operators such as add, delete and modify to create a feature representation. The feature representation maintains additional information such as feature names, taxonomy codes and attributes that are not kept in a conventional solid modeller and this eliminates the need for feature recognition. The functional requirements of a feature based design system were summarized by Pratt and Wilson (1985) and Shah et al (1988b), as being: a) the data supported must be sufficient for all applications which will use the database; b) The mechanism for feature definitions must be flexible (generic) to allow designers to define features in any form, at any level for their own needs. c) the product definition system must provide an attractive environment for creating, manipulating, modifying and deleting feature entities. Feature relationships should also be defined. d) The design system should be able to integrate with different application software and the interface mechanism should be flexible or generic so that the effort to integrate with different software can be minimized.

Shah and Rogers (1988, 1989) introduced a Feature Based Modelling System, which consists of a feature based modelling shell (integrated with a CSG based solid modeller), and a feature mapping shell. The feature modelling shell provides all necessary facilities for creating a product database except the actual definition of features. The solid representation of form features are stored as a Feature Producing Volumes (CSG sub-trees) and Boolean operators. Feature operations such as modifying, deleting and manipulating are available since the CSG syntax is applied. A wireframe co-modeller has been incorporated to provide quick response to the user when interacting with the model. The feature mapping shell is provided for the purpose of extracting and reformulating product data as needed by the specific application. Since the feature definitions are different for different applications, the feature mapping shell should be generic, so that the integration of the feature modelling shell with different applications is made easier. Pratt and Wilson (1985) studied the representation and manipulation of features in a geometric solid model. They suggested that application programs should manage their own data unless the data has been formally specified by the modeller as being part of the modeller's data. A feature processor has been advocated to manipulate feature data and communicate with the geometric modeller. A feature model database is suggested for the exchange of information with the geometric model database. The Applications Interface Specification (AIS) is intended to allow the user to interact with both the geometric modeller and the feature processor.

Chang (1989) and Turner and Anderson (1988) developed a system, called QTC (Quick Turnaround Cell) which is linked to, but not dependent on, the TWIN BRep based solid modeller. A design by features user interface generates the design data, in the form of feature lists which are sent to the process planning system. The planning system reasons about the exact manufacturing features, their relationships and feasible tool approach directions. This reasoning process is called feature refinement, and uses both the feature lists from the design system and the BRep model re-created in the planning system to produce a complete manufacturing feature model for a part.

Dixon (1988) proposed a knowledge based design system, which is not associated with a solid modeller. The system consists of two parts. The first part consists of a user interface, a design by features library, an operations library (add, modify, and delete) and a monitor (which ensures that the operations requested and performed by the user are allowable and understandable to the system). These allow the user to create primary representations of features. The second part of the system is used for converting the primary representations into secondary representations where all the information needed for subsequent reasoning must be available.

Chung et al (1988) developed a knowledge based design system which is used in the domain of the investment casting process. The overall system consists of a user interface system, a knowledge based reasoning system, a solid modeller, and a communication interface between the solid modeller and the reasoner. In the prototype system, the user interacts with the reasoning system through the user interface, whilst the solid modeller runs in the background. The reasoning system parses the user's commands and sends requests to the solid modeller through the communication interface for geometry manipulation and geometric data query. The solid modeller then sends back the information required by the reasoning system which builds the feature data model for the object. The difference between Chung's and Dixon's systems is that a solid modeller is used in Chung's system which increases the capability of handling geometric entities. However, in Chung's system, the user works directly with the knowledge based system, and this limits the flexibility of the design by features front end, because of the limited facilities offered by the knowledge based system. For example, only the 'add' feature operation is provided.

5.3 Comparison of the Two Approaches

Although the feature recognition approach allows the design of parts using existing conventional CAD systems which have sophisticated geometric modelling facilities, there are some severe problems. In particular, only a limited number of features can be recognized from a solid model, the pattern matching process is complicated especially for 3D complex objects and the definition of features is not precise (the same geometry may be converted into different features by different processing algorithms).

The design by features approach can eliminate the need for feature recognition and gives a unique, pre-defined feature list with which designers may construct their parts. However, the systems at present still impose limitations on designers: the design by features library is finite, and the feature operations, such as add, delete and edit are frequently limited. The flexibility and freedom of designing the geometry of an arbitrary part in conventional CAD systems have been lost to some degree in the design by features systems, although Li et al (1991) and Requicha and Vandenbrande (1989) have tried to overcome this problem.

6. Conclusions

The information in a geometric model of a part designed using conventional CAD systems is not sufficient for process planning or other reasoning purposes. Hence, a feature model needs to be created which contains not only the geometric and topological information, but also all other required information such as feature taxonomy codes, tolerances, surface finishes, materials, and tool access directions. There are two main alternatives for constructing this feature model, the feature recognition approach and the design by features approach.

cd deThe feature recognition process is complex, the number of features that can be recognized is limited, and the designer's intent is lost. The design by features approach allows the designer to model a part in terms of features, and this eliminates the need for feature recognition whilst maintaining design intent.

The main problems with the design by features approach are limitations on the feature library, and unavailability of certain feature operations (such as add, edit, delete and transform) in current tested systems, loss of design flexibility, and the lack of a general application independent and relatively stable form feature database. Such a database could minimize the feature mapping (or feature conversion) process.

The solid modeller used should be based on CSG methods, with boundary representations generated when required. Such a hybrid CSG/BRep approach combines the advantages of both CSG and BRep modelling methods. The hybrid approach also allows a feature to be defined as a volumetric feature, without losing the ability of attaching dimensions and tolerances data to the faces, edges and vertices of the feature.

The application of Features technology can be said to be a mature research activity, but at present only limited evidence of commercial exploitation is available. There are commercially available geometric modelling systems which are either based on feature descriptions or make such descriptions available as an optional method of working. Unfortunately, such feature interaction is limited to model creation and plays no part in downstream activities such as design analysis and process planning. In part this lack of exploitation of features may be attributed to the inconclusive nature of the research findings. More research is required to resolve the divisions in the research community regarding the use of design by features or feature recognition, a unifying taxonomy suitable for a wide range of applications, standardisation on an international scale and demonstrations of the usefulness of a features representation throughout the product life-cycle.

7. References

ALTING, L., and ZHANG, H., 1989, Computer aided process planning: the state-of-the-art survey, *International Journal of Production Research*, **27**, 4, 553-585.

BRIMSON, J. A., and DOWNEY, P. J., 1986, Feature technology: a key to manufacturing integration, *CIM review*, Spring 1986, 21-27.

BROOKS, S. L., HUMMEL, K. E., and WOLF, M. L., 1987, XCUT: a rule-based expert system for the automated process planning of machined parts, *Proceedings, Symposium on Integrated and Intelligent Manufacturing*, ASME Winter Annual Meeting, Boston, MA, USA.

BUTTERFIELD, W. R., GREEN, M. K., SCOTT, D. C., and STOKER, W. J., 1985, Part features for process planning, *C-85-PPP-03*, CAM-I Inc., Arlington, Texas, USA.

CASE, K., and ACAR, B. S., 1989, The manufacturing features approach to design and its impact on the learning and use of computer aided design systems. In E. D. Megaw (ed.) *Contemporary Ergonomics*, (Taylor and Francis, London), 132-137.

CHANG, T. C., 1989, *Expert Process Planning For Manufacturing.* (Addison-Wesley, Reading, MA, USA).

CHOI, B. K., BARASH, M. M., and ANDERSON, D. C., 1984, Automatic recognition of machined surfaces from a 3D solid model. *Computer-Aided Design*, **16**, 2, 81-86.

CHUANG, S. H., and HENDERSON, M. R., 1990, Three-dimensional shape pattern recognition using vertex classification and vertex-edge graphs. *Computer-Aided Design*. **22,** 6, 377-387.

CHUNG, J. C. H., COOK, R. L., PATAL, D., and SIMMONS, M. K.. 1988, Feature-based geometry construction for geometric reasoning. *Proceedings of the 1988 ASME International Computers in Engineering Conference and Exhibition,* San Francisco, CA, USA, **1**.

CUNNINGHAM, J. J., and DIXON, J. R., 1988, Designing with features: the origin of features. *Proceedings of the 1988 ASME International Computers in Engineering Conference and Exhibition*, San Francisco, CA, USA, **1**, 237-243.

DIXON, J. R.,. 1988, Designing with features: building manufacturing knowledge into more intelligent CAD systems. *ASME Manufacturing International '88*, Atlanta GA, USA, 51-57.

FAUX, I. D., 1986, Reconciliation of design and manufacturing requirements for product description data using functional primitive part features. *R-86-ANC/GM/PP-01.1*., CAM-I Inc. Arlington, Texas, USA.

FROST, R. A., 1986, *Introduction to Knowledge Based Systems.* (Collins Professional and Technical Books, London).

GANDHI, A., and MYKLEBUST, A., 1989, *A Natural Language Approach to Feature Based Modelling.* (Mobil Research and Development Corporation, Princeton, New Jersey, USA).

GAO, X., and CASE, K., 1991, *Process capability modelling: A review report of feature representation methodologies.* (Department of Manufacturing Engineering, Loughborough University of Technology, UK*).*

GAVANKAR, P., and HENDERSON, M. R., 1990, Graph-based extraction of protrusions and depressions from boundary representations. *Computer-Aided Design*. **22**, 7, 442-450.

GINDY, N. N. Z., 1989., A hierarchical structure for form features. *International Journal of Production Research*. **27**, 12, 2089-2103.

GOSSARD, D. C., ZUFFANTE, R. P., and SAKURAI, H., 1988, Representing dimensions, tolerances, and features in MCAE systems. *IEEE Computer Graphics & Applications*, March 1988, 51-59.

HART, N. M., 1986, A CAD engineering language to aid manufacture. *PhD Thesis* (Department of Manufacturing Engineering, Loughborough University of Technology, UK).

HENDERSON, M. R., 1986, Automated group technology part coding from a three-dimensional CAD database. In S. C. Lu and R. Komanduri (eds) *Knowledge-Based Expert System for Manufacturing,* (ASME), PED-VOL24,195-204.

HENDERSON, M. R., CHUANG, S. H., GANU, P., and GAVANKAR, P., 1990, *Graph-Based Feature Extraction* (Arizona State University, *USA).*

HERBERT, P. J., HINDE, C. J., BRAY, A. D., LAUNDERS, V. A., ROUND, D., and TEMPLE, D. M., 1990, Feature recognition within a truth maintained process planning system. *International Journal of Computer Integrated Manufacturing*. **3,** 2, 121-132.

HINDE, C. J., BRAY, A. D., HERBERT, P. J., LAUNDERS, V. A., and ROUND, D., 1990, An integrated approach to process planning. *International CIS Journal,* 42-60.

VAN HOUTEN, F. J. A. M., VAN T'ERVE, A. H., and KALS, H. J. J., 1989, PART, a feature based computer aided process planning system. *SPIN-project PART*. (University of Twente, The Netherlands), Report No PA89-008.

HUMMEL, K. E., and BROOKS, S. L., 1986, Symbolic representation of manufacturing features for an automated process planning system. In S. C. Lu and R. Komanduri (eds) *Knowledge-Based Expert System for Manufacturing,* (ASME), PED-VOL24, 233-243.

JARED, G. E. M., 1986, Feature recognition and expert systems for operation planning in NC machining. *Proceedings of the International Conference on AI Europa*, Wiesbaden, Germany.

JARED, G. E. M., 1989a, *The feature recognition battle - latest from the front.* (Department of Applied Computing and Mathematics, Cranfield Institute of Technology, UK).

JARED, G. E. M., 1989b, Reasoning and using geometric features. In J. Woodwark (ed.) *Geometric Reasoning* (Clarendon Press), 169-184.

JARED, G. E. M., 1991, Geometric Reasoning for Economic Manufacture. *Progress report on SERC Grant No GR/F/70785.*

JOSEPH, A. T., KALTA, M., and DAVIES, B. J., 1990, Automatic generation of NC programs for turned components from CAD product models. *Proceedings of the 28th MATADOR*, 289-294.

JOSHI, S., and CHANG, T.C., 1988, Graph-based heuristics for recognition of machined features from a 3D solid model. *Computer-Aided Design*. **20**, 2, 58-66.

JOSHI, S., and CHANG, T. C., 1990, Feature extraction and feature based design approaches in the development of design interface for process planning. *Journal of Intelligent Manufacturing.* 1, 1-15.

LEE, Y. C., and FU, K. S., 1987, Machine understanding of CSG: extraction and unification of manufacturing features. *IEEE Computer Graphics & Applications*, 7(1), 20-32.

LEE, K. I., LEE, J. W., and Lee, J. M., 1989, Pattern recognition and process planning prismatic workpieces by knowledge based approach. *Annals of the CIRP*. **38**. 1, 485-488.

LI, R. K., and YU, M. H., 1990, A framework for prismatic part-data generation - unit-machine loop concept. *International Journal of Computer Integrated Manufacturing*. **3**. 2, 96-111.

LI, R. K., TAUR, B. W., and SHYUR, H. J. 1991, A two-stage feature-based design system. *International Journal of Production Research*. **29**. 1, 133-154.

LUBY, S. C., DIXON, J. R., and SIMMONS, M. K., 1986, Designing with features: creating and using a features data base for evaluation of manufacturability of castings. *Computers in Mechanical Engineering (ASME)*, **5**(3), 25-33.

MORTENSEN, K. S., and BELNAP, B. K., 1989, A rule-based approach employing feature recognition for engineering graphics characterisation. *Computer-Aided Engineering Journal*, December, 221-228.

NNAJI, B. O., and LIU, H. C., 1990, Feature reasoning for automatic robotic assembly and machining in polyhedral representation. *International Journal of Production Research*. **28**. 3, 517-540.

PERNG, D. B., CHEN, Z., and LI, R. K., 1990, Automatic 3D machining feature extraction from 3D CSG solid input. *Computer-Aided Design*. **22**. 5, 285-295.

PRATT, M. J., and WILSON, P. R., 1985, Requirements for support of form features in a solid modelling system. *R-85-ASPP-01*. CAM-I Inc., Arlington, Texas, USA.

PRATT, M. J., 1988, Synthesis of an optimal approach to form feature modelling. *Proceedings of the 1988 ASME International Computers in Engineering Conference and Exhibition*, San Francisco, CA, USA, **1**, 263-274.

REQUICHA, A. A. G., and VANDENBRANDE, J. H., 1989, Form feature for mechanical design and manufacturing. *Proceedings of the 1988 ASME International Computers in Engineering Conference and Exhibition*, Anaheim, CA, USA, August 1989, **1**, 47-52.

REQUICHA, A. A. G., 1980, Representations for rigid solids: theory, methods and systems. *ACM Computing Surveys*. **12.** 4. 437-464.

REQUICHA, A. A. G., 1990, Solid modelling and its applications: progress in tolerancing, inspection, and feature recognition. *Proceedings of the 16th NSF Design & Manufacturing Systems Grantees Conference*, 1-12.

ROY, U., and LIU, C. R., 1988, Feature-based representational scheme of a solid modeler for providing dimensioning and tolerancing information. *Robotics and Computer-Integrated Manufacturing*. **4**, (3/4), 335-345.

SHAH, J. J., and ROGERS, M. T., 1988, Functional requirements and conceptual design of the feature-based modelling system. *Computer-Aided Engineering Journal*, February 1988, 9-15.

SHAH, J. J., BHATNAGAR, A., and HSIAO, D., 1988a. Feature mapping and application shell. *Computer Aided Engineering*, 489-496.

SHAH, J. J., SREEVALSAN, P., ROGERS, M. T., BILLO, R., and MATHEW, A.. 1988b (Published in 1989). Current status of features technology. *CAM-I Report R-88-GM-04.1*.

SHAH, J. J., and ROGERS, M. T.,. 1989, Geometric modeling for automated manufacturing. *Proceedings of the 16th NSF Design & Manufacturing Systems Grantees Conference*, 371-375.

SHAH, J. J., and ROGERS, M. T., 1990, Feature based modelling shell: design and implementation. *Computers in Engineering Journal*, January, 255-261.

TURNER, G. P., and ANDERSON, D. C., 1988, *An Object-Oriented Approach to Interactive, Feature-Based Design for Quick Turnaround Manufacturing.* Report No TR-ERC88-6 (Schools of Engineering, Purdue University, West Lafayette, Indiana, USA).

VARVAKIS, G. J., 1991, Manufacturing code generation for rotational parts in a feature-based product modelling environment. *PhD Thesis* (Department of Manufacturing Engineering, Loughbrough University of Technology, UK).

WANG, H. P., and WYSK, R. A., 1987, Intelligent reasoning for process planning. *Computers in Industry*. **8**, 293-304.

WANG, H. P., 1987, A knowledge based computer aided process planning system. *Knowledge based expert systems in engineering: planning and design*. Computational Mechanics Publication, 259-269.

WANG, H. P., and WYSK, R. A., 1988, AIMSI: a prelude to a new generation of integrated CADCAM systems. *International Journal of Production Research*. **26**. 1, 119-131.

WOODWARK, J. R., 1988, Some speculations on feature recognition. *UKSC Report 186* (IBM, Winchester, UK).