In Kosuke Ishii, editor, *ASME Computers in Engineering Conference*, pages 93–104. ASME, September 1994

FEATURE RECOGNITION FOR MANUFACTURABILITY ANALYSIS

William C. Regli
Dept. of Computer Science and
Institute for Systems Research
University of Maryland

Satyandra K. Gupta
Mechanical Engr. Dept. and
Institute for Systems Research
University of Maryland

Dana S. Nau
Dept. of Computer Science and
Institute for Systems Research
University of Maryland

ABSTRACT

While automated recognition of features has been attempted for a wide range of applications, no single existing approach possesses the functionality required to perform manufacturability analysis. In this paper, we present a methodology for taking a CAD model of a part and extracting a set of machinable features that contains the complete set of alternative interpretations of the part as collections of MRSEVs (Material Removal Shape Element Volumes, a STEP-based library of machining features). The approach handles a variety of features including those describing holes, pockets, slots, and chamfering and filleting operations. In addition, the approach considers accessibility constraints for these features, has an worst-case algorithmic time complexity quadratic in the number of solid modeling operations, and modifies features recognized to account for available tooling and produce more realistic volumes for manufacturability analysis.

1. INTRODUCTION

Automated recognition of features has been attempted for a wide range of applications. Although many of the CAD/CAM applications addressed in previous approaches have had compatible goals and functionality, it is often unclear what specific classes of parts, features, and feature interactions can be handled, making it difficult to evaluate their utility for manufacturability analysis.

This paper describes a method for representing the alternative interpretations of a part as a set of machining features and translating its CAD model into a set of features guaranteed to contain all the alternatives. The context for this work is manufacturability analysis of machined parts, in which consideration of alternative feature interpretations is necessary to produce realistic and reliable feedback to the designer.

Although several approaches have previously been developed for recognizing features from CAD models, there are several new issues addressed by our work:

Standards. For purposes of integrating CAD with CAM, it is important to be able to get features that correspond directly to manufacturing operations. Moreover, standard schemes have not been employed for representing features, therefore existing systems cannot be directly used by many commercially available computer aided manufacturing applications.

To address this problem, we use a class of features that are expressible as MRSEVs (Material Removal Shape Element Volumes) [17]. MRSEVs are volumetric features corresponding to machining operations on 3-axis milling machines. MRSEVs can be defined using EXPRESS (the official STEP information modeling language) and STEP form features. By employing a set of features based on STEP, we have attempted to address a domain of machinable parts and features of wide interest and facilitate interface with STEP-based CAM tools.

Representing Alternatives. In general, there may be several alternative interpretations of the part as different collections of machinable features, each corresponding to different ways to machine the part. Determining which of these alternatives is most preferable requires considering the part dimensions, tolerances, and surface finishes, the availability and capabilities of machine tools and tooling, and fixturability constraints. Many approaches try to generate a single interpretation for a given part—but in general, there may be many alternative interpretations of the part, each of which many need to be generated and examined to evaluate the manufacturability of the part or determine an optimal plan.

To address this, our approach represents the alternative feature-based interpretations for the part and defines the feature recognition problem independent of the implementation of the algorithm used to solve it.

Recognition of Alternatives. While many approaches for recognizing features exist, it is often difficult to determine the extent to which changes in feature topology and geometry due to their mutual intersections can be handled. An implementation may work well for some types of parts but be unable to address more complex parts and interactions. Hence, it is sometimes unclear what specific classes of parts and feature interactions can be handled and which alternative feature-based interpretations of the part can be produced by existing approaches.

To address this, we have developed a methodology for finding all of the features in a mathematically specifiable class of alternative feature-based models directly from a solid model of an arbitrary part and piece of stock. This class of machinable parts is described by MRSEV features, and includes parts described by features that intersect with each other in complex ways. The subclass of the MRSEV library we consider include hole, pocket, and edge-cut features, along with accessibility constraints for those features. The algorithm's worst-case time complexity is quadratic in the number of solid modeling operations.

In our previous work [9, 25, 26], we had focused on developing a formalization of the problem of recognizing machinable features expressible as MRSEVs and demonstrating provable completeness and complexity properties for our algorithms. This paper builds on these results, emphasizing the link between MRSEVs and machining operations, their relationship to evolving standards for data exchange, and some of the unique characteristics of our approach that address some of the requirements of manufacturability analysis.

This paper is organized as follows: Section 2 discusses related work. Section 3 presents relevant definitions and feature subclasses. Section 4 presents an overview of our approach for recognizing features and its implementation. Section 5 describes the application of this feature recognition methodology to manufacturability analysis. Section 6 discusses future extensions of our research and concluding remarks.

2. RELATED WORK

Feature-based approaches have been very popular in a variety of CAD/CAM implementations. Significant amounts of work have been directed towards defining sets of form or machining features to serve as a communication medium between design and manufacturing—but at present, most researchers are convinced that there is no single set of features that satisfy the requirements of both of these domains. Most relevant to automated manufacturability analysis is recent work on feature recognition and generation of alternative feature interpretation for parts. For a comprehensive overview of feature-based manufacturing techniques, the reader is referred to [29].

Feature recognition has been considered an important research area in CAD/CAM integration and many different approaches have been developed over the last decade. The approaches of [4, 13] based on graph algorithms have known algorithmic properties but do not difficult to extend more complex feature types. Both grammatical methods and some graph-based approaches are prone to combinatorial difficulties [21]. The work in [6] describes promising techniques that combat the combinatorial problems by abstracting an approximation of the geometric and topological information in a solid model and finding features in the approximation. Corney and Clark [3] have had success extending the capabilities of graph-based algorithms to more general $2\frac{1}{2}$-dimensional parts.

In one of the early efforts on feature extraction, Woo [33] proposed a method for finding general depression and protrusion features on a part through decomposing the convex hull of the solid model. The approach had several problems, including the existence of pathological cases in which the procedure would not converge. The non-convergence of Woo's approach has been solved in recent work by Kim [14]. Currently being addressed is how to extend this method from polyhedra to handle the types of curved surfaces found in realistic parts. In recent work, Sakurai and Chin [28] propose an approach for recognizing general protrusions and cavities through "spatial decomposition and composition."

The work of Henderson [11] was seminal in employing expert systems on the feature recognition problem. More recently, Henderson has employed graph-based methods and neural networks to recognize features [2, 22]. Kyprianou [18] presented the first effort to use grammars to parse solid models of parts for group coding.

The ability to handle interacting features has become an informal benchmark for feature recognition systems and has been the focus of numerous research efforts. The work of [5] included the formalization of a feature description language and employed frame-based reasoning algorithms to extract machining features for computer aided process planning. An aggressive approach to handle feature interactions and intersections was done by Marefat [20]. The work built on the representation scheme of [13] and used a novel combination of expert system and hypothesis testing techniques to extract surface features from polyhedral objects.

A comprehensive approach for recognizing features and handling their interactions is that of Vandenbrande [32]. Employing recognition "hints" for each feature class, hints are extracted from the solid model and classified as to their potential for building a feature instance. A frame-based reasoning system then acts on the hints and attempts to complete a feature frame with information needed to make a maximal instance of a feature.

The recent work of Laakko and Mäntylä [19] couples feature-based design and feature recognition to provide for incremental feature recognition. This type of approach recognizes changes in the geometric model as new or modified features while preserving the existing feature information. They also provide for some forms of customizability with use of a feature-definition language to add new features into the system.

3. DEFINITIONS AND NOTATION

3.1. Basic Concepts

A *solid* is a manifold r-set with analytic bounding surfaces. If R is any solid, then $b(R)$ is the *boundary* of R. If R and R' are solids, then $R \cap^* R'$ is the *regularized intersection* of R and R'. Similarly, $R \cup^* R'$ and $R -^* R'$ are the *regularized union* and *regularized difference*, respectively [27].

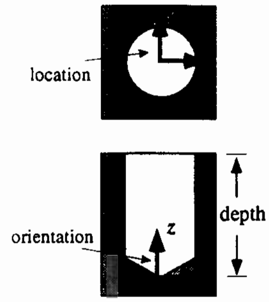
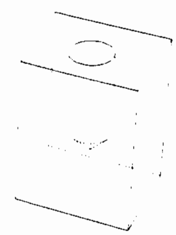
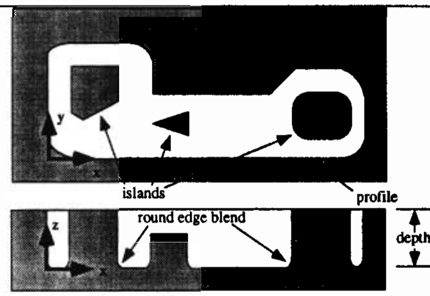
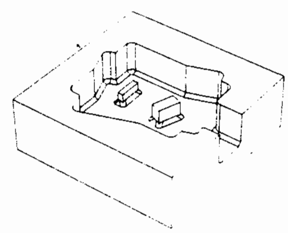
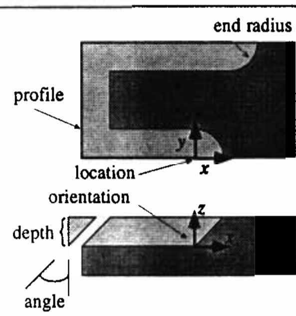
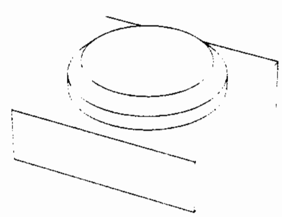
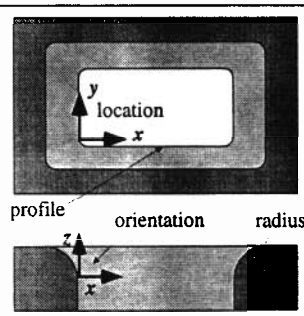
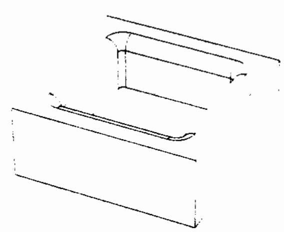
Attributes	Description	Example
Hole Feature **location** p **radius** r **orientation** $\vec{v}$ **depth** d		
Pocket Feature **location** p **orientation** $\vec{v}$ **depth** d **profile** E **bottom blend** b_t, b_d **islands** I		
Edge-Flat Feature **location** p **orientation** $\vec{v}$ **profile** E **depth** d **angle** a **end radius** r_e		
Edge-Round Feature **location** p **orientation** $\vec{v}$ **radius** r **profile** E **end radius** r_e		

Figure 1: Subclasses of MRSEV machining features.

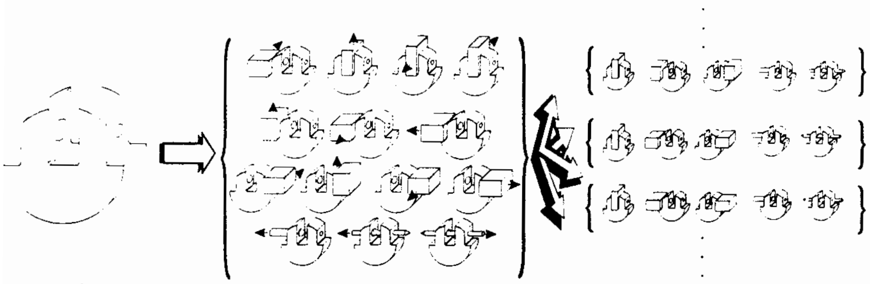


Figure 2: Feature-based models.

A *machined part* (or just a *part*) is the finished component to be produced as a result of a finite set of machining operations on a piece of *stock*, i.e., the raw material from which the part is to be machined. We will represent both the part and the stock as geometric solids. We use term *workpiece* to describe the state of stock after applying a subset of operation sequences. Throughout this paper, we let P be a solid representing a part, and S be a solid representing the stock from which P is to be made. The *delta volume* (i.e., the volume to be machined), is the solid $\Delta = S -^* P$.

A feature has conditions that must be met in order for it to be machinable. A feature f is *valid* for a given part P, if:

1. f creates some portion of the boundary of P (i.e., $b(f) \cap b(P) \neq \emptyset$);
2. f does not intersect P (i.e., $f \cap^* P = \emptyset$);
3. f is accessible.

We represent accessibility as a solid associated with f representing the volume occupied by the non-cutting portion of the machine tool as it moves during the machining operation. A detailed presentation of accessibility is beyond the scope of this paper.

We define each alternative interpretation of the part as a set of machining features to be a feature-based model of the part.

Feature-Based Models. Let P be the given part and S be the given stock. We define a *feature-based model* (FBM) of P and S to be a finite set of features instances M having the following properties:

1. If we subtract the features in M from S, we get P; i.e., $S -^* \cup^*_{f \in M} f = P$.
2. No feature in M is redundant, i.e., for every feature $f \in M$, $S - \cup^*_{f' \in M - \{f\}} f' \neq P$.

Intuitively, a feature-based model represents a single interpretation of the delta volume as a set of machining features, as shown in Figure 2.

We will be interested in features that correspond to the the maximal realistic machinable volume made by a single machining operation in a single machining setup.

Primary Features. A *primary* feature for a part P and stock S is any valid feature f, that satisfies the following conditions:

1. Every valid feature f' that contains f (having the same location and orientation and made with the same machining operation) also has the same effective material volume as f (i.e., if $f \subset f'$ then $f' \cap^* S = f \cap^* S$).
2. Every valid feature f' that is contained in f (having the same location and orientation and made with the same machining operation) has a smaller effective material volume (i.e., if $f' \subset f$ then $f' \cap^* S \subset f \cap^* S$).

Primary features represent largest realistic feature instance capable of machining the required volume, any larger feature would include extra volume outside that of the workpiece. During generation of an operation sequence, a primary feature can be truncated to obtain the feature instance that will be employed by the operation plan that produces the geometry. Generation of operation plans is discussed in Section 5.

3.2. Material Removal Shape Element Volumes (MRSEVs)

3.2.1 PDES/STEP. STEP is the International *St*andard for the *E*xchange of *P*roduct Model Data being developed by the International Organization for Standardization (ISO). PDES stands for *P*roduct *D*ata *E*xchange using *S*TEP and it represents the activity of several organizations in the United States in support of STEP. The organizations involved with PDES comprise many corporate, government, and standards development entities.

Describing data in STEP is handled by defining an information model in the EXPRESS data modeling language [31] for each type of data required. Once an information model is defined, data for representing a specific product can be represented by using the STEP rules for mapping EXPRESS to a physical file [1]. The EXPRESS model defines the data entities that describe the class of objects in the domain.

3.2.2. The MRSEV Hierarchy. Kramer [17] developed a library of material removal shape element volumes (MRSEVs) as a means of categorizing the shapes of volumes to be removed by machining operations on a 3-axis machining center, such as drilling and milling. MRSEVs can be defined using the EXPRESS modeling language and STEP form features, the details of which are presented in [17, 16].

The MRSEV hierarchy provides a framework for describing a large class of volumetric entities of interest to machining. Each entity type has a number of required and optional attributes. MRSEV instances have been used for applications such as process planning and NC-program generation [15]. Kramer's primary MRSEV types include linear swept features, edge-cut features, ramps, general grooves, and rotational pockets.

MRSEVs are geometrically and topologically defined volumetric features. Information about design attributes such as tolerances, surface finishes, available machine tools, or specific operations to machine the volumes are not part of their definitions. While the CAD models we consider may have specification for such attributes, selection of the appropriate operations is not done within the scope of our feature recognition system. In the context of this approach, consideration of these types of machining constraints and choices for specific operations that machine a MRSEV volume is performed during the manufacturability analysis, as will be discussed in Section 5. The operations used to machine a MRSEV will depend on the cost and availability of tools and machines to satisfy these design attributes and the parameters considered to analyze manufacturability.

For the purpose of this paper we confine our domain to the edge-cut and linear swept features, i.e., holes, pockets, and pockets with islands, chamfers, fillets, countersinks, and edge blends. Kramer defines linear swept feature as a shape resulting from sweeping a closed profile of edges along a straight line perpendicular to the plane of the profile. In the case of a pocket with islands, an island is considered to be a subfeature defined by its own closed profile. A bottom blend on a pocket is a transition surface between a pocket's bottom face and its walls. An edge-cut feature results from sweeping the flat or round edge of an angled tool along a, possibly open, profile of edges. The product of this kind of MRSEV feature is typically a flattened or rounded edge on the part, such as a chamfer. We have incorporated manufacturability criteria (such as accessibility and existence of corner radii for convex pocket corners) to MRSEVs to ensure the volumes recognized are in some way machinable. The table in Figure 1 presents illustrations for the subclasses of MRSEV feature we will address in this paper: MRSEV holes, pockets, edge flats, and edge rounds.

3.3. Correspondence: Machining Operations and MRSEVs

To perform a machining operation, one starts out with a rotating cutting tool. The cutting tool is mounted on a large machine tool, and the total volume occupied by the cutting tool and the machine tool is quite large. But we will only be interested in some small portion of this total volume, namely the portion that actually gets close to the workpiece. This *tool volume* has a boundary consisting of both cutting and non-cutting surfaces. To perform the machining operation, one sweeps the tool volume along some trajectory. Informally, the solid consisting of the set of points hit by the cutting surfaces of the tool as it is swept along the trajectory will be the material removed by a machining operation.

MRSEVs can be used to represent volumes which can be removed during machining. For example, a MRSEV hole represents a volume which can be removed by a drilling operation, and a MRSEV pocket represents a volume which can be removed by an end or face milling operation. It is worth noting that the "pocket" MRSEV is used not only to represent what is usually called a pocket, but also to represent a large variety of milled shapes such as steps, profiles, slabs, etc.

4. RECOGNIZING FEATURES

Given solids representing the part P and the stock S as input, we are interested in finding the set of all features that may be used to determine machining operations of an operation plan to create P. This raises several questions as to which feature instances are to be addressed, which features get recognized, and which feature-based models can be produced. Ideally, a feature recognition algorithm is *complete* if, for all P and S, if it returns the set of all valid features that can be used in produce P from S. This is an unrealistic expectation, because for any given P and S there may be, theoretically speaking, an infinite number of possible valid feature instances. Even restricted to primary features, cases arise where a simple machinable parts can be modeled by infinitely many primary features. We will instead consider completeness with respect to the *well-behaved* features, as defined below:

Well-Behaved Features A feature f is *well-behaved* if it is primary and f satisfies any of the following properties:

1. f is a hole MRSEV and f subsumes a hole MRSEV f' with a subface of one of its side or ending surfaces contained by $b(\Delta)$.
2. f is a pocket MRSEV and f subsumes a pocket MRSEV f' for which there is a pair of non-colinear coplanar edges, e_1 and e_2, from the set of edges bounding the faces of Δ and on $b(f')$ where the plane defined by e_1 and e_2 contains or, in the event of a through feature, is parallel to the bottom surface of f'.
3. f is an edge-cut MRSEV and f subsumes an edge-cut MRSEV f' whose edge-cutting operation produces a set of connected surfaces contained by $b(\Delta)$.

An FBM is well-behaved if every feature f in the FBM is well-behaved. For a part P, we denote that the set of all well-behaved feature instances as $\mathcal{F}$.

4.1 Feature Recognition Procedures

We can now present an outline for algorithms to construct the set $\mathcal{F}$ of all well-behaved features from an arbitrary part P and stock S. Our approach uses solid modeling operations basic to boundary-representation solid modeling systems and proceeds from the observation that each geometric attribute of the delta volume must be created by some feature instance f. The basic idea is to traverse the geometric

```
BUILD_F(P, S)
INPUT: solid models of a part P and stock S
OUTPUT: a set of well-behaved feature instances, F.
        Initially, F = ∅.

    for all geometric attributes
       g of Δ do
      // geometric attributes considered
      // are edges and faces
      F = F ⋃
        CONSTRUCT_HOLES(g, P, S) ⋃
        CONSTRUCT_POCKETS(g, P, S) ⋃
        CONSTRUCT_EDGE-FLATS(g, P, S) ⋃
        CONSTRUCT_EDGE-ROUNDS(g, P, S)
    end for
    return(F)
```

Figure 3: A high-level description of the algorithm for constructing $\mathcal{F}$.

information of the delta volume and instantiate the primary MRSEV features capable of covering all or a portion of each surface. A high-level description of the algorithm is given in Fig. 3.

A feature instance is a parameterized solid. For each MRSEV subclass, we have an algorithm for constructing a primary feature instance from a given set of attribute values and verifying its accessibility constraints:

NEW_HOLE_MRSEV($p, r, \vec{v}, d, e$)
NEW_POCKET_MRSEV($p, \vec{v}, d, E, I$)
NEW_EDGE-FLAT_MRSEV($p, a, \vec{v}, d, E, r_e$)
NEW_EDGE-ROUND_MRSEV($p, r, \vec{v}, d, E, c, s, r_e$)

If no such feature instance can be constructed (i.e. for example, the parameterization gives rise to a feature or an accessibility volume with a non-empty intersection with the part P), then the functions report so.

Space does not permit elaboration on each of the functions for constructing primary features from part and stock geometry and topology. We present below an outline of these algorithms, some of which have appeared in greater detail in [9, 26]. Implementation of these will vary depending on the functionality of the modeling system chosen.

1. CONSTRUCT_HOLES: Hole MRSEVs are perhaps easiest to recognize. An instance of a hole can be found from their end surface face or their cylindrical side face. In the case of a cylindrical face made by hole that extending through the part, there are two possible primary feature instances: one in each direction along the axis of the cylindrical surface. For non-through hole features, only one feature instance exists and it can be constructed either from the end face or from the side face using algorithms that have been given previously in [9, 25].

2. CONSTRUCT_POCKETS: Construction of pocket features starts at an edge of the part, e_1. For each edge e_1 in the part, the faces meeting at e_1 have the potential of belonging to three different types of feature instance:

```
CONSTRUCT_EDGE-FLATS(g, P, S)
INPUT: solid models of a part P and stock S,
      a geometric attribute g.
OUTPUT: a set of well-behaved feature instances, F

    determine if g is a surface creatable
      with an edge-flat operation
    if yes:
     for each available tool tip angle do
      determine possible orientations for
          tool to machine g: v1, v2
      calculate depths of possible feature
          instances: d1, d2
      find possible edge profiles from
          surfaces connected to g: E1, E2
      find end radii: r_e1, r_e2
      build feature instances: f1, f2
      add features to F
     end for
    return(F)
```

Figure 4: Outline of edge-flat procedure.

(a) As pictured in Figure 5(a), edge e_1 could be an edge of one of the bottom surface of a pocket instance. For example, e_1 could belong to the planar bottom face or be part of a blend surface.

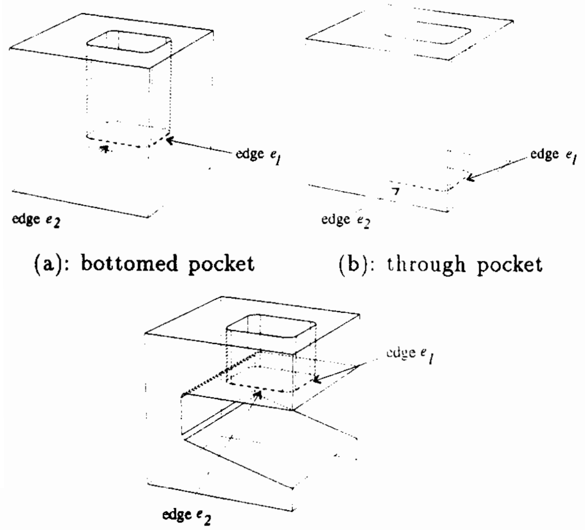
(b) As pictured in Figure 5(b), edge e_1 could be an edge of a side surface of a pocket instance which extends through the part. This type of feature is would be a through pocket.

(c) As pictured in Figure 5(c), edge e_1 could be an edge of a side surface of a pocket instance having no bottom surface present in the part.

The orientation of the pocket feature is determined from the edges e_1 and e_2. In the event of a through pocket, there are two possible orientations.

The profile for the pocket can be computed from the projection of the part faces that lie in the half-space above (with respect to the orientation) the plane containing the bottom surface of the feature. The projection is computed onto the the plane containing the endpoints of the edges e_1 and e_2. In first and third cases, this plane is the bottom surface of a pocket that creating the edges and their adjacent surface. Note that this applies when the bottom surface of the pocket has been eliminated through interactions with other features.

In the second case, the feature extends through the part and thus has no bottom surface present in the delta volume. For this situation, e_1 and e_2 provide the orientation for the through pocket and an arbitrary location can be chosen for the projection plane. The



(a): bottomed pocket (b): through pocket

(c): bottomless pocket

Cases possible pocket features.

aces are mapped onto it from both the direction of the normal and its opposite, arriving at a cross-section of the through pocket that might have created these edges.

3. CONSTRUCT_EDGE-FLATS and CONSTRUCT_EDGE-ROUNDS: Given a face belonging to the contiguous profile of surfaces made by an edge-cutting operation as shown in Figure 6(a), we construct a feature instance as follows:
   (a) Determine the possible orientations for feature instances that may have machined the surface. In the case of chamfering features, there may be a several possible setups, depending on which tool angle we choose to make the surface.
   (b) Determine the full extent of the feature by examining adjacent part edges and faces in directions perpendicular to the orientation with similar geometries (i.e. in the case of chamfers, the same dimensions; in the case of fillets, the same radii). Figure 6(b) indicates the orientations for 45 degree chamfering tool.
   (c) Construct the feature instances, as shown in Figure 6(c).

An outline of the edge-flat procedure is shown in Figure 6. The procedure for edge-round MRSEVs is similar.

We have shown [25, 24] that our feature recognition procedures are complete with respect to well-behaved features, i.e. for any part P and stock S, the algorithms produce all well-behaved feature instances that can be used to model

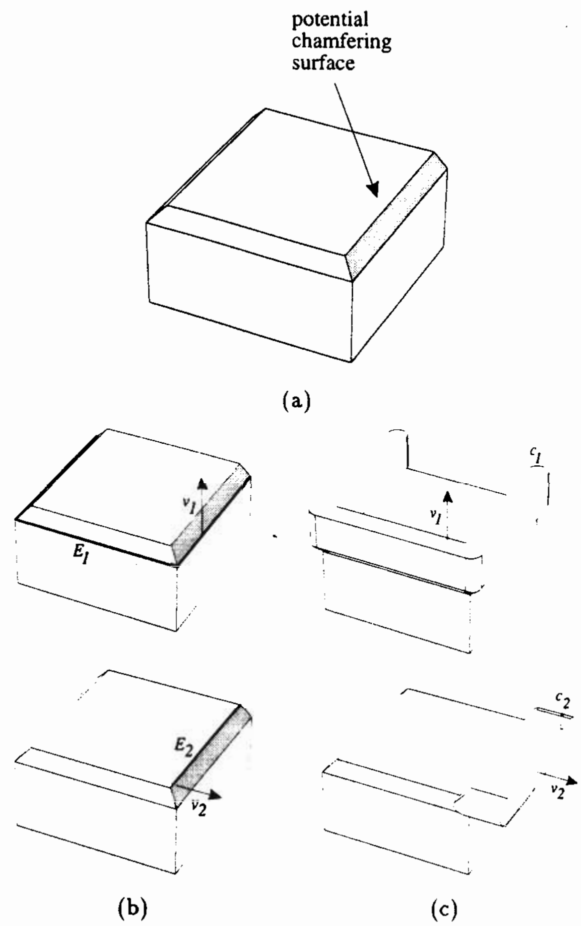


Figure 6: Recognizing edge-flat features.

P and S. In this way, the set $\mathcal{F}$ produced can be used to generate all of the alternative FBMs for P and S.

4.2. Profile Offsetting

The edge profile of a MRSEV pocket may present unreasonable constraints on the selection of tooling and operations. For instance, as calculated above the profile may contain convex corners that cannot be machined by a milling tool. Further, in some cases the most cost cost-effective way to machine a MRSEV pocket is to perform the machining operation using the largest possible tool. Such situations may require that the tool moves outside the boundary of the stock material. To take these machinability considerations into account, we offset the profiles of pocket MRSEVs.

Given an instance of a pocket MRSEV, its profile contains two types of edges: (1) *closed edges:* those which belong to pocket faces whose exterior's (with respect to the volume of the pocket) are incident with some portion of the surface of the part and (2) *open edges:* those belonging to pocket faces whose exterior's are incident exclusively with a portion of the surface of the stock.

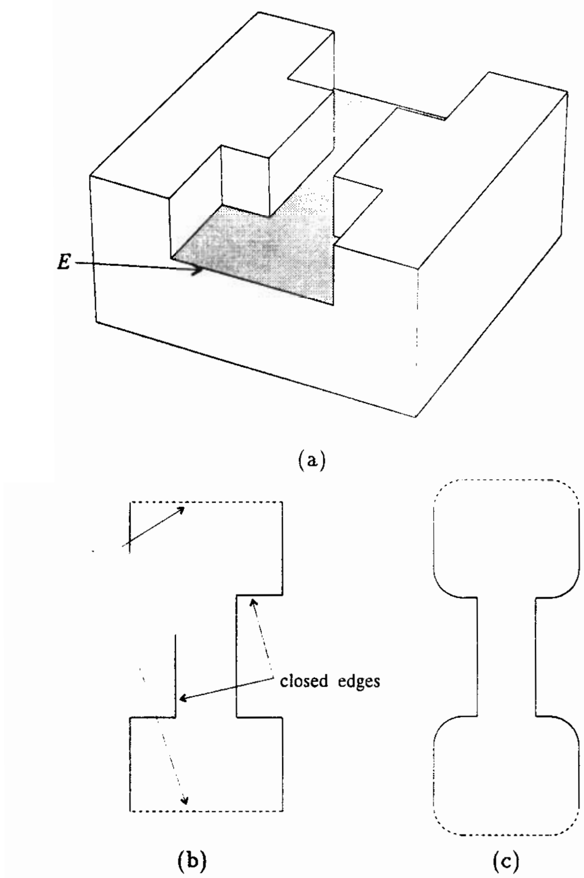


Figure 7: An example of profile offsetting.

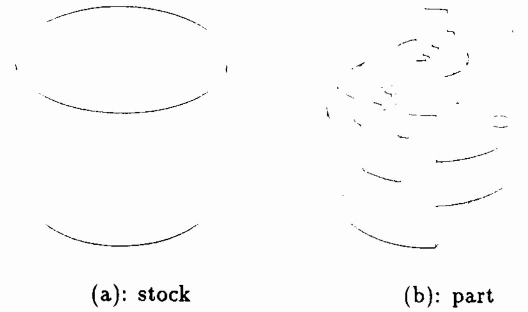


Figure 8: An example part and stock.

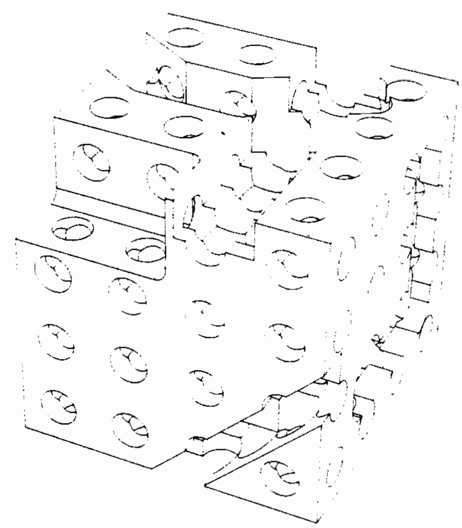


Figure 9: A part with a variety of feature interactions.

The steps in the offsetting procedure are as follows:

1. **Estimation of an optimal tool size.** In a typical milling operation, a larger tool diameter implies a shorter cutting trajectory and less operation cost and time. However, a variety of constraints resulting from the geometric configuration of the profile will restrict the maximum tool size that may be employed. In this step, the geometry of the profile is used to calculate an upper bound on the tool size.
2. **Alteration of the profile.** In some profiles, the estimation of tool size may reveal machinability problems. For example, two adjacent closed profile edges meeting at a convex corner results in a tool radius estimate of zero; a narrow distance between closed edges in the profile may return an estimate smaller than the smallest available tool. This step modifies profiles by offsetting convex corners inward to account for the corner radius left by a tool or dividing up an otherwise unmachinable profile into a set of multiple ones that can be machined with available tooling.
3. **Offsetting of the profile.** After finding usable bound on the tool size, the open edges of a MRSEV pocket are found are offset outward by the tool radius in order to allow the tool to move on or outside of these edges during machining.

Figure 7 provides an illustration of the feature offsetting process, a detailed presentation of which appears in [24, 8]. In the figure, the edges of profile E have been offset to take into account the radius of a cutting tool.

4.3. Examples.

Figure 8 shows an example part. If this part is machined from a cylindrical piece of stock, Figure 10 shows the various MRSEV features identified by our algorithms. In this case, feature set $\mathcal{F}$ consists of those features within the box.

Figure 9 illustrates a particularly intricate solid with a variety of volumetric feature interactions. While itself not a particularly useful component, it does present some aspects of the complexities to be found in realistic parts where feature geometry and topology may become highly disconnected or distorted. This example has 346 part faces and

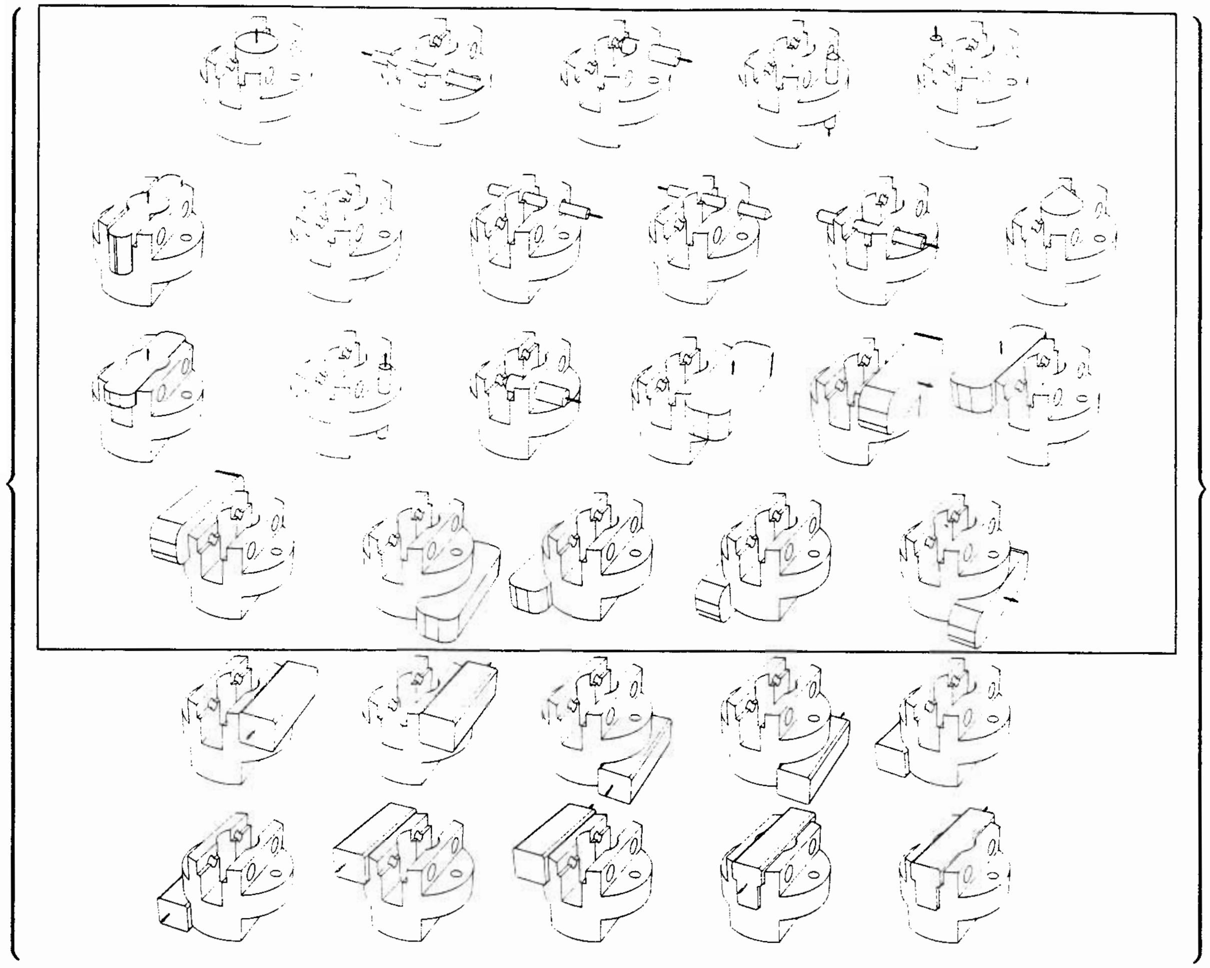


Figure 10: MRSEVs identified by our algorithm for the part shown in Figure 8. The feature-based models of the part are comprised of the features within the box. Due to machining considerations, no features outside the box contribute to an FBM.

413 delta volume faces; considered with a rectangular block of stock material, yields 106 drilling and twelve milling features. The twelve milling features are shown in Figure 11.

4.4. Implementation

We have built a proof-of-concept implementation of this feature recognition methodology in C++ using version 3.0.1 of the AT&T C++ compiler from SUN Microsystems, version 1.5.1 of Spatial Technologies' ACIS® solid modeling kernel, and the NIH C++ Class Library developed at the National Institutes of Health. Also being employed in our development efforts are Ithaca Software's HOOPS® Graphics System and the Tcl/Tk embeddable command language and user interface toolkit developed at the University of California at Berkeley.

The current implementation of the feature recognizer omits bottom blends on pockets as they are not crucial to the downstream applications. Implementation for general through pockets is restricted by the current version of the ACIS® application procedural interface. At the time of this writing, we are extending the ACIS® application procedural interface to provide the needed functionality.

5. GENERATING AND EVALUATING FEATURE-BASED MODELS FROM A FEATURE SET

Since the main focus if this paper has been to describe a methodology for finding all features that are elements of feature-based models, we will not discuss the specifics of these applications. Rather our goal in this section is to outline the potential CAM applications that exploit the properties of this feature recognition methodology. For more information on the specifics of these applications, readers are referred to [9, 10].

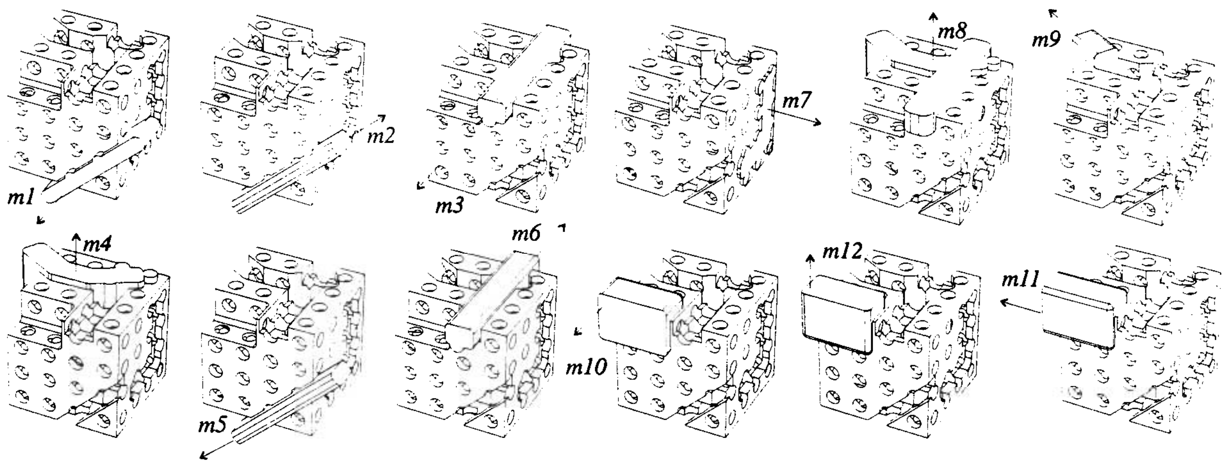


Figure 11: Twelve pocket features from Figure 9.

After finding the feature set $\mathcal{F}$, the next step is to use these features to generate feature-based models for the design. Many times, $\mathcal{F}$ contains redundant features (i.e., same portion of the delta volume is covered by more than one feature). For most CAM applications, we will be interested in feature-based models of the part: collections of features having no redundant elements (i.e., we don't want to machine the same volume twice) as defined in Section 3. In general, for a given part there may be more than one feature-based model, each one corresponding to a potential way of making the part. Each feature-based model is basically an irredundant set cover of the delta volume from $\mathcal{F}$, models can be generated using variations on irredundant-set-covering techniques[23], and use pruning heuristics to discard unpromising models.

Consider an evaluation function which estimates the manufacturability of a given feature-based model. In most of the cases, we are interested in finding the feature-based model which optimizes this evaluation function. For example, an evaluation function might use the feature-based model to consider production cost, production time or other factors related to manufacturability. Guaranteeing the optimality of a the solution might depend on the consideration of the complete class of feature-based models of the part. In practice this may not always be feasible, however an analysis of completeness can produce information useful for determining the limitations of the system and identifying the potential sources for problems when they do occur.

Besides optimizing the evaluation function value, a feature-based model may need to satisfy additional constraints. For example, in case of process planning, operations associated with the feature-based model should be capable of meeting the tolerance requirements. Moreover, for a feature-based model to be useful for process planning, there must exist a sequence of machining operations such that during all stages of machining, the intermediate workpiece geometry is suitable for fixturing and setup. Given a candidate operation sequence, the machining data for that sequence, the feature's dimensions, and the material from which the part is to be made, we can evaluate whether or not it can satisfactorily achieve the tolerance specifications. As there may be many ways to achieve these constraints, it is important that the feature recognizer produce a satisfactory set for evaluation of all of these alternatives.

6. CONCLUSIONS

We have described our approach for recognition of machining features from CAD models for the purposes of manufacturability analysis. The algorithms we present take a CAD model and extract all instances of MRSEV features occurring in any of the alternative feature-based models for the given part. We have proven the approach to be complete over a significant class of parts.

Some of the primary characteristics of our approach are as follows:

1. While various CAD and CAM applications may have compatible goals and functionality, their specific details have been different enough that integration has proven difficult. To address this, our approach recognizes features from the MRSEV library, describing general machining operations on 3-axis machining centers. In addition to feature recognition, these algorithms can be viewed as a means of translation from a solid model to a STEP representation.
2. Our approach handles a variety of hole, pocket, and edge-cut MRSEVs. When possible, the MRSEV features recognized are offset to account for the dimensions of the milling tool to be used. The new offset fea-

tures correspond more naturally to the area which will be machined to produce the desired removal volume. This allows us to simplify our calculation of machining cost and time by eliminating complex tooling profiles given by the pocket features constructed solely from the part and the delta volume.

3. The feature recognition algorithm is complete over a class of realistic parts, even if the features intersect with each other in complex ways. Knowing the limits on completeness is useful for applications such as manufacturability analysis, in which determining the manufacturability of a design may require trying many alternative FBMs to see which one is best.
4. The algorithm's time complexity is quadratic in the number of solid modeling operations.

Future goals include continuing to enhance our implementation, extending our results and procedures to include other MRSEV features, and exploring techniques to reduce the computational costs. Further, we hope to exploit the object-oriented structure to the MRSEV hierarchy to support extensibility and user-defined features. We are currently considering how to use an object-oriented paradigm to generalize the results and algorithms to other feature hierarchies and user-defined feature classes.

ACKNOWLEDGEMENTS

This work was supported in part by NSF Grants NSFD CDR-88003012, DDM-9201779, and by the University General Research Board.

[1] J. Altemueller. Mapping from EXPRESS to physical file structure. Technical Report ISO TC184/SC4 Document N280, International Organization for Standardization, 1988. September.

[2] S. H. Chuang and M. R. Henderson. Three-dimensional shape pattern recognition using vertex classification and the vertex-edge graph. *Computer Aided Design*, 22(6):377–387, June 1990.

[3] J. Corney and D. E. R. Clark. Method for finding holes and pockets that connect multiple faces in $2\frac{1}{2}$d objects. *Computer Aided Design*, 23(10):658–668, December 1991.

[4] Leila De Floriani. Feature extraction from boundary models of three-dimensional objects. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(8), August 1989.

[5] Xin Dong. *Geometric Feature Extraction for Computer-Aided Process Planning.* PhD thesis, Rensselaer Polytechnic Institute, Troy, NY, USA, 1988.

[6] R. Gadh and F. B. Prinz. Recognition of geometric forms using the differential depth filter. *Computer Aided Design*, 24(11):583–598, November 1992.

[7] N.N.Z. Gindy. A hierarchical structure for form features. *Int. J. Prod. Res.*, 27(12):2089–2103, 1989.

[8] Satyandra K. Gupta. *Automated Manufacturability Analysis of Machined Parts.* PhD thesis, The University of Maryland, College Park, MD, August 1994.

[9] 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*, 1994. To appear.

[10] S.K. Gupta and D.S. Nau. A systematic approach for analyzing the manufacturability of machined parts. *Computer Aided Design*, 1994. To appear.

[11] Mark R. Henderson. *Extraction of Feature Information from Three-Dimensional CAD Data.* PhD thesis, Purdue University, West Lafayette, IN, USA, 1984.

[12] K. E. Hummel. The role of features in computer-aided process planning. In *Proceedings of Feature Symposium*, Woburn, Boston, MA, August 1990.

[13] S. Joshi and T. C. Chang. Graph-based heuristics for recognition of machined features from a 3D solid model. *Computer-Aided Design*, 20(2):58–66, March 1988.

[14] Y. S. Kim. Recognition of form features using convex decomposition. *Computer Aided Design*, 24(9):461–476, September 1992.

[15] Thomas R. Kramer. An express schema for machining plugged into ALPS4. Technical report, The National Institute of Standards and Technology, Gaithersburg, MD 20899, January 1991.

[16] Thomas R. Kramer. Issues regarding material removal shape element volumes (MRSEVs). Technical Report NISTIR 4804, The National Institute of Standards and Technology, Gaithersburg, MD 20899, March 1992.

[17] Thomas R. Kramer. A library of material removal shape element volumes (MRSEVs). Technical Report NISTIR 4809, The National Institute of Standards and Technology, Gaithersburg, MD 20899, March 1992.

[18] Lycourgos K. Kyprianou. *Shape Classification in Computer Aided Design.* PhD thesis, Christ College, University of Cambridge, Cambridge, United Kingdom, 1980.

[19] Timo Laakko and Martti Mäntylä. Feature modelling by incremental feature recognition. *Computer Aided Design*, 25(8):479–492, August 1993.

[20] M. Marefat and R. L. Kashyap. Geometric reasoning for recognition of three-dimensional object features. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(10):949–965, October 1990.

[21] Thomas J. Peters. Mechanical design heuristics to reduce the combinatorial complexity for feature recognition. *Research In Engineering Design*, 4:195–201, 1993.

[22] S. Prabhakar and M. R. Henderson. Automatic form-feature recognition using neural-network-based techniques on boundary representations of solid models. *Computer Aided Design*, 24(7):381–393, July 1992.

[23] J. A. Reggia, D. S. Nau, and P. Y. Wang. A formal model of diagnostic inference. II. algorithmic solution and applications. *Information Sciences*, 37:257–285, 1985.

[24] William C. Regli. PhD thesis, The University of Maryland, College Park, MD, 1994. In preparation.

[25] William C. Regli and Dana S. Nau. Building a general approach to feature recognition of material removal shape element volumes (MRSEVs). In Jaroslaw Rossignac and Joshua Turner, editors, *Second Symposium on Solid Modeling Foundations and CAD/CAM Applications*, New York, NY 10036, USA, May 19-21, Montreal, Canada 1993. ACM SIGGRAPH, ACM Press.

[26] William C. Regli and Dana S. Nau. Recognition of volumetric features from CAD models: Problem formalization and algorithms. Technical Report TR 93-41, The University of Maryland, Institute for Systems Research, College Park, MD 20742, USA, April 1993.

[27] Aristides A. G. Requicha. Representation for rigid solids: Theory, methods, and systems. *Computing Surveys*, 12(4):437–464, December 1980.

[28] Hiroshi Sakurai and Chia-Wei Chin. Defining and recognizing cavity and protrusion by volumes. In Fatih Kinoglu, editor, *ASME Computers in Engineering Conference*, pages 59–65, August 1993.

[29] Jami Shah, Martti Mäntylä, and Dana Nau, editors. *Advances in Feature Based Manufacturing*. Elsevier/North Holland, 1993.

[30] Jami J. Shah. Assessment of features technology. *Computer Aided Design*, 23(5):331–343, June 1991.

[31] P. Spiby. Exchange of product model data – part 11: The EXPRESS language. Technical Report ISO TC184/SC4 Document N14, International Organization for Standardization, 1991. April.

[32] J. H. Vandenbrande and A. A. G. Requicha. Spatial reasoning for the automatic recognition of machinable features in solid models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(12):1269, December 1993.

[33] Tony C. Woo. Feature extraction by volume decomposition. In *Conference on CAD/CAM Technology in Mechanical Engineering*, pages 76–94, March 1982.