Computer Aided Design, 1994, to appear.

A Systematic Approach for Analyzing the Manufacturability of Machined Parts

Satyandra K. Gupta* Dana S. Nau†

University of Maryland
College Park, MD 20742 USA

Abstract

The ability to quickly introduce new quality products is a decisive factor in capturing market share. Because of pressing demands to reduce lead time, analyzing the manufacturability of the proposed design has become an important step in the design stage. This paper presents an approach for analyzing the manufacturability of machined parts.

Evaluating the manufacturability of a proposed design involves determining whether or not it is manufacturable with a given set of manufacturing operations—and if so, then finding the associated manufacturing efficiency. Since there can be several different ways to manufacture a proposed design, this requires us to consider different ways to manufacture it, in order to determine which one best meets the design and manufacturing objectives.

The first step in our approach is to identify all machining operations which can potentially be used to create the given design. Using these operations, we generate different operation plans for machining the part. Each time we generate a new operation plan, we examine whether it can produce the desired shape and tolerances, and calculate its manufacturability rating. If no operation plan can be found that is capable of producing the design, then the given design is considered unmachinable; otherwise, the manufacturability rating for the design is the rating of the best operation plan.

We anticipate that by providing feedback about possible problems with the design, this work will help in speeding up the evaluation of new product designs in order to decide how or whether to manufacture them. Such a capability will be useful in responding quickly to changing demands and opportunities in the marketplace.

*Institute for Systems Research and Mechanical Engineering Department. Email: skgupta@src.umd.edu
†Institute for Systems Research and Department of Computer Science. Email: nau@cs.umd.edu

1 Introduction

In today's competitive market, the ability to quickly introduce new quality products is a decisive factor in capturing market share. Because of pressing demands to reduce lead time, increasing research attention is being given to integration of engineering design and manufacturing. These attempts have led to the evolution of the *design for manufacturability* (DFM) methodology [2]. This methodology involves simultaneously considering design goals and manufacturing constraints in order to identify and alleviate manufacturing problems while the product is being designed, thereby reducing the lead time and improving the product quality.

Even as early as World War II [28], efforts have been made to implement DFM methodology. Traditional approaches range from building inter-departmental design teams to providing designers with DFM checklists. Recently, attempts have been made to automate some aspects of DFM [2, 9, 12, 13, 23]; most of these attempts involve off-line manufacturability evaluation of designs using rule-based systems. The advent of sophisticated CAD/CAM system has provided the opportunity of tighter integration of CAD and DFM. This will require embedding manufacturing guidelines and capabilities in CAD systems, in order to analyze the manufacturability of the proposed design.

The main objective of our work is to develop a tool for computer-aided DFM, that can be used during early design stages to improve the product quality from the manufacturing point of view. We expect that such a tool will enable the designers to build manufacturability into a design through analysis of its shape, dimensions and tolerances with respect to given manufacturing processes. Our tool is intended to be similar to other design-analysis tools (such as FEA, mechanism analysis, etc.)—except that our tool will analyze and report problems with manufacturability, rather than functionality.

In this paper, we present our approach to automated manufacturability analysis of machined parts. In our analysis, we are interested in determining whether or not a proposed design is manufacturable with a given set of manufacturing operations—and if so, then finding the associated manufacturing efficiency.

Evaluating manufacturability involves finding a way to manufacture the proposed design, and estimating the associated production cost and quality. However, there often can be several different ways to manufacture a proposed design—so to evaluate its manufacturability, we need to consider different ways to manufacture it, to determine which one is best. In our approach, we systematically generate and evaluate alternative operation plans, to see which ones can produce the design to the desired specifications—and we estimate the production cost and time of each plan, to measure its manufacturing efficiency.

We expect that the information provided by our approach will be useful in providing feedback to the designer about possible problems that may arise in trying to meet the specified geometry and tolerances. We hope this will allow the designers to correct the manufacturing problems during the design stage, thereby producing the designs that will be easier to manufacture.

2 Overview of the Approach

In a typical CAD environment, the designer creates a design using solid-modeling software, and uses analysis software to examine different aspects of the proposed design's functionality. As shown in Fig. 1, we propose extending the design loop to incorporate a manufacturability analysis system that can be used once the geometry and/or tolerances have been specified. This will help in creating designs that not only satisfy the functional requirements but are also easy to manufacture.

In order to analyze the manufacturability, we need information about the proposed design and available manufacturing resources. In our approach, we assume that the proposed design is available as a solid model, along with the tolerance and surface finish information as attributes of various faces

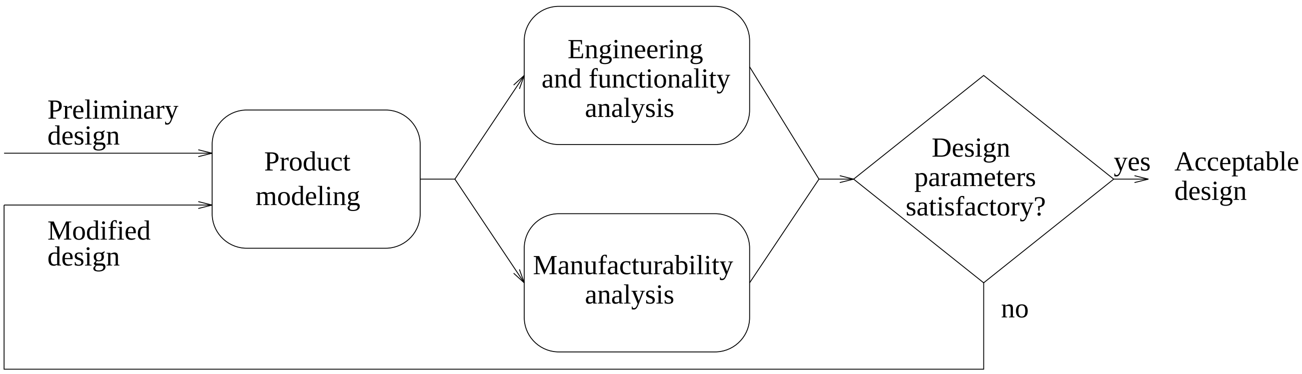


Figure 1: Our proposed extension to a typical design loop.

of the solid model. Furthermore, we assume that we have information about the available machining operations—information that includes the process capabilities, dimensional constraints etc. In Section 4 we explain how this information can be modeled using machining features.

The basic idea behind our approach is to generate alternative interpretations of the part as collections of machining features, map these interpretations into operation plans, and evaluate the manufacturability of each operation plans, as shown below:

Step 1. As described in Section 5, generate the set of all machining features that can be used to create the design. Each feature in this set represents a different possible machining operation which can be used to create various surfaces of the part. If this set of features is not sufficient to machine the design completely, then the design is not machinable, so exit with failure. Otherwise, proceed to the next step.

Step 2. As described in Section 6, do the following steps repeatedly, until every promising feature-based model (FBM) for the design has been examined. An FBM is basically a set of machining features that contains no redundant features and is sufficient to create the design.

- **Step 2a.** As described in Section 6, generate a promising FBM from the set of machining features generated in Step 1. We consider an FBM unpromising if it is not expected to result in any operation plans better than the ones that have already been examined.
- **Step 2b.** As described in Section 7, generate the set of promising operation plans from the FBM generated in Step 2a. Each operation plan represents a partially ordered set of machining operations. We consider an operation plan to be unpromising if it violates any common machining practices.
- **Step 2c.** As described in Section 8, estimate the achievable machining accuracy of each operation plan generated in Step 2b. This determines which of the operation plans are capable of producing the design tolerances and surface finishes.
- **Step 2d.** As described in Section 9, calculate the manufacturability rating of each operation plan that is capable of producing the design tolerances and surface finishes. The manufacturability rating of an operation plan is based on production time and cost estimates.

Step 3. If one or more operation plans were found during the above steps that are capable of producing the desired tolerances and surface finishes, then the design is machinable, so return the best manufacturability rating. Otherwise, the design is not machinable.

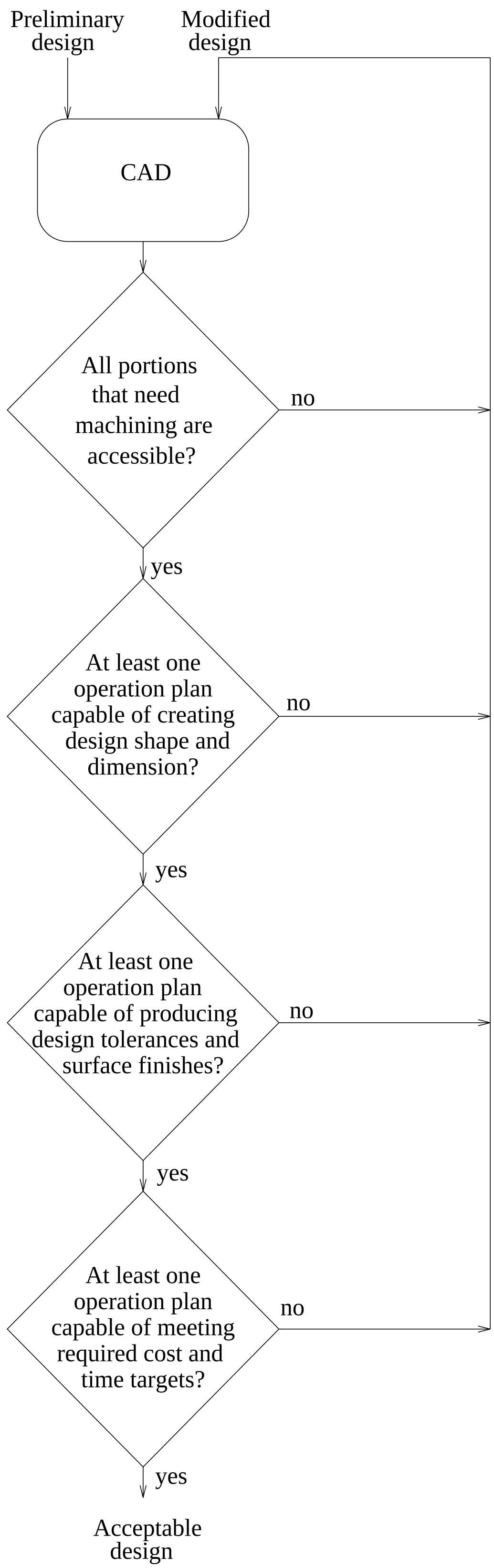


Figure 2: Manufacturability analysis.

As shown in Figure 2, there are four cases in which the above approach will find that the design is not machinable. In these cases, the designer will need to consider making changes to the design, as described below:

1. If some portions of the design do not correspond to any machining features, then these portions should be modified or eliminated.

2. If no operation plan can be found that is capable of creating the design shape and dimensions, then the dimensions and/or shape are unsatisfactory and should be changed such that at least one operation plan works.

3. If no operation plan can satisfy the design tolerances, then designer should consider loosening the unachievable tolerances or change the design shape and/or dimensions.

4. If there are specified time or cost targets and no operation plan is capable of meeting those targets, then designer should consider changing the design to eliminate those design characteristics that require expensive or time-consuming machining operations.

Fig. 3 shows some example parts handled by our approach. The part shown in Fig. 3(a) is unmachinable, because the four notches on the corner of the part do not have any corner radiuses. The part shown in Fig. 3(b) is a modified version of the part, which is machinable. The part shown in Fig. 3(c) is machinable, but is expensive to machine because it requires a large number of setups. Fig. 3(d) shows a modified version of the part, that requires fewer setups to machine and therefore has better manufacturability.

Our approach can also be used incrementally as a design involves. In particular, the steps that involve operation planning (without estimating machining accuracy) can be used to analyze the design shape and dimensions of the design. After tolerances has been specified, the steps involving estimation of machining accuracy can be used to analyze the design tolerances.

3 Related Work

3.1 Feature Extraction

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 based on graph algorithms (such as [15]) and alternating sum-of-volume decomposition (such as [17]) have known algorithmic properties, but appear difficult to extend to realistic manufacturing problems. Grammatical methods and some of the graph-based approaches are prone to combinatorial difficulties. The recent work in [9] 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.

Perhaps the most comprehensive and formal approach to date for recognizing features and handling their interactions has been that of Vandenbrande [24]. Their method is capable of coming up with alternative feature interpretations and is described in Section 3.2.

The absence of a clear mathematical formalism for the problem has made it difficult to ensure completeness of these approaches. In particular, when features intersect with each other, this changes their topology and geometry in ways not taken into account by most existing feature recognition systems. Hence, it is often unclear what specific classes of parts and feature interactions can be handled by various existing approaches. Our approach (see Section 5) is intended to address this problem.

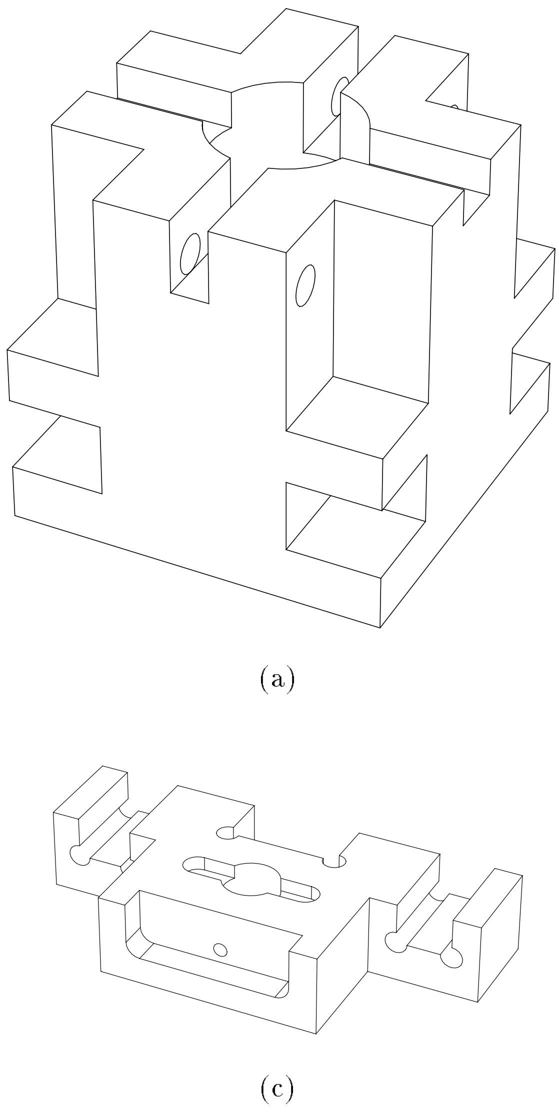


Figure 3: Some of the example parts handled by our approach.

3.2 Generation of Alternatives

For the purposes of manufacturability analysis it is very important to examine alternative ways of manufacturing the part—and this requires obtain alternative interpretations of the part as different collections of machinable features. Our approach to this problem is described in Section 6; below is a summary of other related approaches and how they differ from ours.

The AMPS process planning system [5] includes a "feature refinement" step, in which heuristic techniques are used for combining a set of features into a more complex feature if it appears that this will optimize the plan, or splitting a feature that cannot be machined into two or more features that can (hopefully) be machined. Since the techniques are heuristic in nature, it is not entirely clear when alternative interpretations will be produced.

Mantyla [19] presents a method for generative generative process planning by using relaxed features. For each of the feature types, a set of geometrically similar features is defined. During process planning any of the alternative features can be used. However, this work does not incorporate precedence constraints or tolerances into its reasoning. Moreover, this approach can not handle cases in which the best plan requires a different number of features than the original one.

Vandenbrande [24] has developed a system that combines techniques from artificial intelligence and solid modeling. The program uses hints or clues to identify potential features in the boundary representation of the part. The system is capable of identifying interacting features (e.g., two intersecting slots). This program also produces alternative features in certain cases. This system does not guarantee generating all potential machining features. Moreover, it does not have any way of grouping the features into feature-based models.

The first systematic work in the direction of generation of alternative interpretations was done by Karinthi and Nau [16]. They described an approach for producing alternative interpretations of the same object as different collections of volumetric features as the result of algebraic operations on the features, and a system for generating alternative interpretations by performing these algebraic operations. However, this system cannot be used directly for manufacturability evaluation, due to the following limitations. First, there was no direct relation between these features and machining operations, so some of the interpretations generated by this approach were not feasible from the machining point of view. Second, the algebraic operators were not sufficient to generate all interpretations of interest for machining purposes. Third, this work did not deal with the time-ordering constraints induced by some kinds of feature interactions.

3.3 Process Planning

A vast literature exists that describes various types of process planning systems [21, 5, 1, 25, 4, 11]. Most of them focus on generating a single plan that is optimal with respect to some criteria. Most of these systems reason with a single interpretations of the part, have very limited geometric reasoning capabilities, and use heuristic techniques To Determine precedence constraints among machining features. Moreover, most do not check to make sure that the intermediate workpiece satisfies the required conditions for proper accessibility. Some of them use complex models for selecting and optimizing various cutting parameters, which makes them too computationally intensive and to be used for manufacturability analysis during the design stage.

For the purpose of manufacturability analysis, we need a planning system that employs geometric reasoning at the planning level to explore the possibility of machining the features in different orders, and determining the status of the current workpiece to assure required conditions for accessibility. Moreover, the planning system should be integrated with the scheme for generating alternative interpretations of the part, to allow pruning of unpromising plans. Our approach (see Section 7) is intended to address these problems.

3.4 Evaluating Manufacturability

Researchers have developed several different approaches to evaluate manufacturability of a given design [2, 9, 12, 13, 23, 8]. Some of these have been developed for specific application domains, while others have been developed for general domains. Most of these approaches are rule-based: design characteristics which improve or degrade the manufacturability are represented as rules, which are applied to a given design in order to estimate its manufacturability. In order to calculate realistic manufacturability ratings, most of these approaches would require large sets of rules.

Research on computer-aided tolerance charting [14, 20] mainly focuses on calculation of optimum intermediate tolerances. The approaches for this task are computationally very intensive, and only consider limited types of tolerances. For the purposes of evaluating design tolerances at early design stages, we instead need a approach that is capable of evaluating the manufacturability aspects of wide variety of tolerances without getting into optimization aspects. Our approach (Section 8) does this.

4 Machining Operations and Machining Features

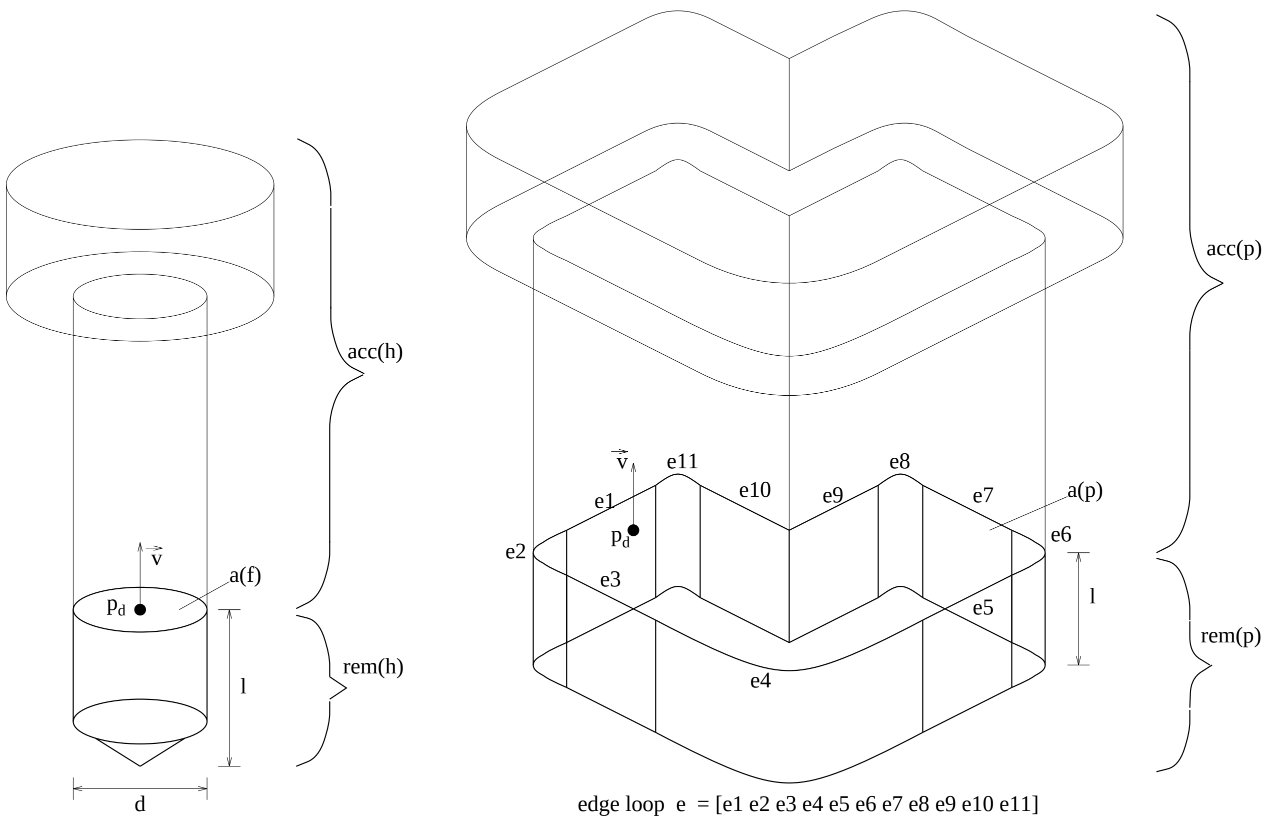
Preliminary definitions and notation. For our purposes, a *solid* is any regular, semi-analytic subset of three-dimensional Euclidean space. If R is any solid, then $b(R)$ is the *boundary* of R, and $\iota(R)$ is the *interior* of R. Note that $R = \iota(R) \cup b(R)$ and that $\iota(R) \cap b(R) = \emptyset$. If R and R' are solids, then $R \cap^* R'$ is the *regularized intersection* of R and R', i.e., the closure of $\iota(R) \cap \iota(R')$. Similarly, $R \cup^* R'$ and $R -^* R'$ are the *regularized union* and *regularized difference*, respectively.

A *part* is the finished component to be produced as a result of a 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. 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*, $\Delta = S -^* P$, is the volume to be machined.

Machining features. In a machining operation, material is removed by relative motion between the cutting tool and the workpiece. For our purposes, a *machining feature* is the portion of the workpiece affected by a machining operation. However, we will need to know not just the volume of material which the feature can remove from the workpiece, but also what kind of machining operation we are performing, how we access the workpiece in order to perform the operation, and so forth.

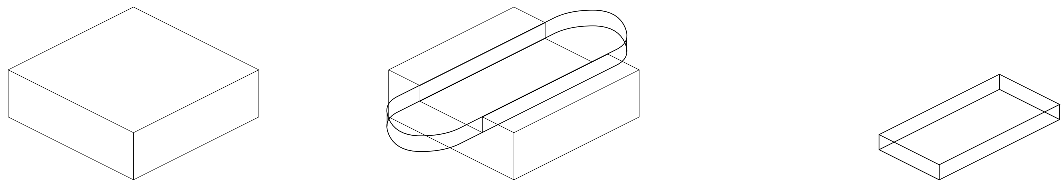
More specifically, a machining feature f will be created by some machining operation $\mathrm{op}(f)$, using a cutting tool $\mathrm{tool}(f)$. To perform the machining operation, one sweeps the tool along some trajectory that is characterized by some set of parameters $\mathrm{param}(f)$. However, only a portion of this swept volume actually corresponds to the volume that can be removed by the machining feature. We refer to this volume as *removal volume* $\mathrm{rem}(f)$. The approach face, $a(f)$, separates the removal volume from the accessibility volume. The *accessibility volume*, $\mathrm{acc}(f)$, is the remaining portion of the tool swept volume. Below are two examples:

- Suppose we want to drill the hole h shown in Figure 4(a). Then $\mathrm{op}(h)$ will be drilling. To create h, we will sweep a drilling tool $\mathrm{tool}(h)$ of diameter d along a linear trajectory starting at the datum point p_d and going in along some unit vector $\vec{v}$ for some distance l. Thus, $\mathrm{param}(h)$ is the set $\{p_d, \vec{v}, d, l\}$. If the approach face $a(h)$ is as shown in the figure, then the removal volume $\mathrm{rem}(h)$ and accessibility volume $\mathrm{acc}(h)$ will be as shown in the figure.



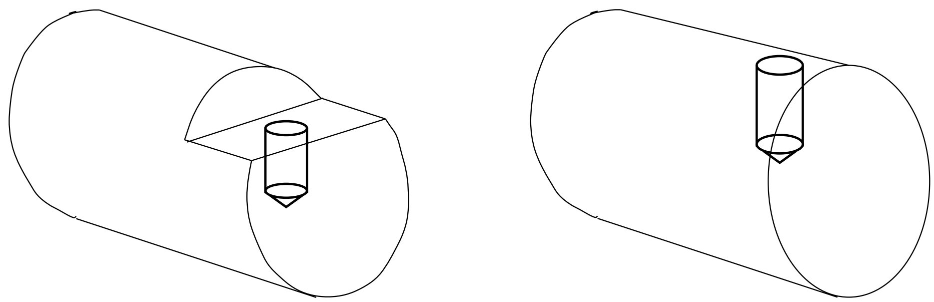
(a): a drilling feature (b): an end-milling feature

Figure 4: Examples of machining features.



(a): the stock (b): a removal volume, and its effect on the stock (c): effective removal volume with respect to the stock

Figure 5: An end-milling feature and its effective removal volume.



(a): the hole is accessible in the workpiece (b): the hole is not accessible because the entry face in not flat.

Figure 6: Examples of feature accessibility.

- Suppose we want to mill the pocket p shown in Figure 4(b). Then op(p) will be milling. To create p, we will sweep an end mill of radius r in plane, whose parameters are the starting point p_d, the depth l, the edge loop e, and the unit orientation vector $\vec{v}$. Thus, param(p) is the set $\{p_d, \vec{v}, e, l\}$. If the approach face $a(p)$ is as shown in the figure, then the removal volume rem(p) and accessibility volume acc(p) will be as shown in the figure.

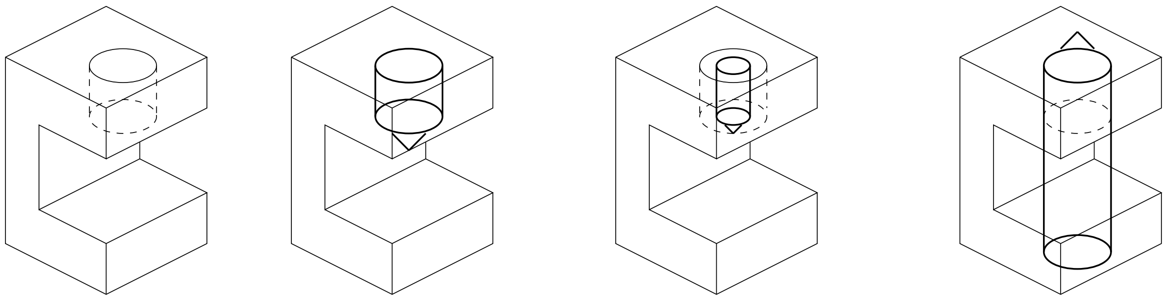
Usually, we will have only a finite set $\mathcal{M}$ of possible machining operations that can be performed; i.e., for each feature f, op(f) must be a member of the set $\mathcal{M}$. Often, we will refer to machining features in terms of the operations used to create them. For example, we say that the hole h is a *drilling feature*, and the pocket p a *end-milling feature*.

Effective removal volume. The volume removed by f from a given workpiece W is not necessarily f's removal volume. Instead, it is f's *effective removal volume with respect to* W, which is defined as $\overline{\text{rem}}(f, W) = W \cap^* \text{rem}(f)$. Fig. 5 shows an end-milling feature and its effective removal volume with respect to the stock.

Accessibility. A feature f is *accessible* in a workpiece W, if the following conditions are satisfied:

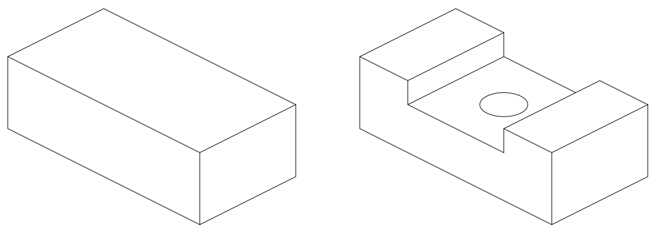
1. The accessibility volume of f does not intersect with W (i.e., $\text{acc}(f) \cap^* W = \emptyset$).
2. The entry face of f should have the proper approach conditions for machining. For example, if the feature f is a drilling feature, then to ensure proper machining, the entry face of the hole f should be a planar surface perpendicular to the hole axis.

Fig. 6 shows examples of accessible and inaccessible features.

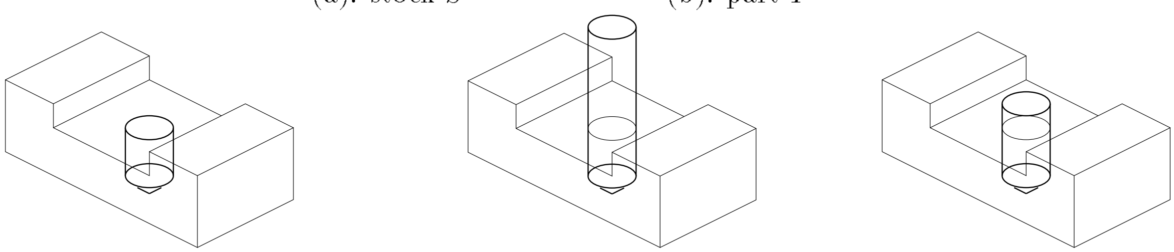


(a): a part (b): a valid feature (c): an invalid feature (d): an invalid feature

Figure 7: Examples of valid and invalid features.



(a): stock S (b): part P



(c): non-primary, because the holes in (d) and (e) have larger effective removal volumes

(d): non-primary, because the hole in (e) has a smaller effective removal volume

(e): primary

Figure 8: Example of primary and non-primary features.

5 Identifying Machining Features

Given solids representing the part P and the stock S, and a set of machining operations $\mathcal{M}$, we are interested in finding the set of all features that correspond to useful machining operations that can be used to create P.

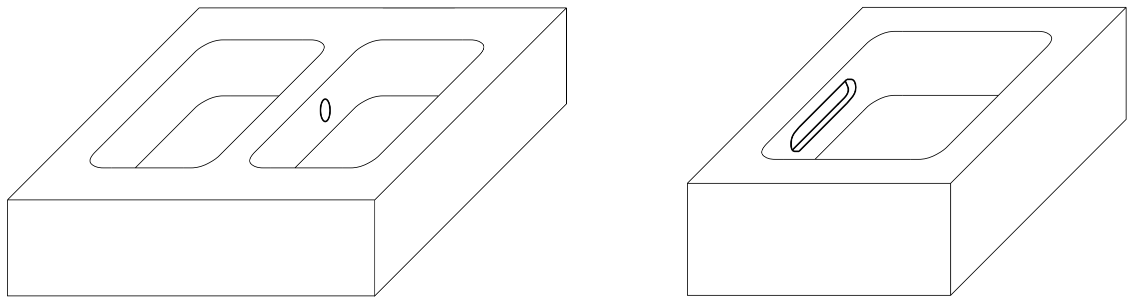
Valid features. A feature f is *valid* for a given part P, if:

1. f creates some portion of the boundary of P (i.e., $b(f) \cap^* b(P) \neq \emptyset$).

2. Removal volume of f does not intersects with P (i.e., $\text{rem}(f) \cap^* P = \emptyset$).

Fig. 7 shows examples of valid and invalid features.

Primary features. A *primary* feature for a part P and stock S is any valid feature f such that the following conditions are satisfied (see Fig. 8 for an example):



(a): the hole on the side wall is not machinable

(b): the slot on the side wall is not machinable

Figure 9: Examples of unmachinable parts, when the set of available machining operations $\mathcal{M}$ is {drilling, end-milling}.

Table 1: Surfaces created by drilling and end-milling features.

Type of feature	Portion of feature boundary	Surface type
hole	bottom	conical (concave)
	side	cylindrical (concave)
end-milling	bottom	planar
	side	cylindrical or planar

1. For every valid feature g (of the same orientation and machining operation as f) whose removal volume contains f's, g has the same effective removal volume as f (i.e., if $\mathrm{rem}(f) \subset \mathrm{rem}(g)$ then $\overline{\mathrm{rem}}(g, S) = \overline{\mathrm{rem}}(f, S)$).
2. For every valid feature g (of the same orientation and machining operation as f) whose removal volume is contained in f's, g has a smaller effective removal volume than f (i.e., if $\mathrm{rem}(g) \subset \mathrm{rem}(f)$ then $\overline{\mathrm{rem}}(g, S) \subset \overline{\mathrm{rem}}(f, S)$).

Algorithm for identifying primary features. Each machining feature is capable of creating certain types of surfaces. For example, Table 1 presents the types of surfaces that can be created by drilling (shown in Fig. 4(a)) and end-milling (shown in Fig. 4(b)) features. In our approach, we consider all the part surfaces that need to be created, and try to identify primary features that are capable of creating those surfaces.

For the details of our algorithm, readers are referred to [22, 10]. It handles a large class of solids composed of features corresponding to drilling and milling operations. The algorithm's time complexity is quadratic in the number of solid modeling operations. Furthermore, the algorithm is provably complete over the set of all solids in our class, even if the features intersect with each other in complex ways.

Example. Suppose we want to design a swivel bracket. Fig. 10 shows two different designs of the bracket; the designs are identical except for the hole diameters. Throughout this paper, we will be using these two designs as examples to illustrate various steps in our approach.

Let us assume that both designs will be machined from a cylindrical stock of carbon steel (100 BHN) with diameter 100mm and length 30mm. We are interested in analyzing the manufacturability of these two designs and selecting the better design. We also assume that both these designs will be

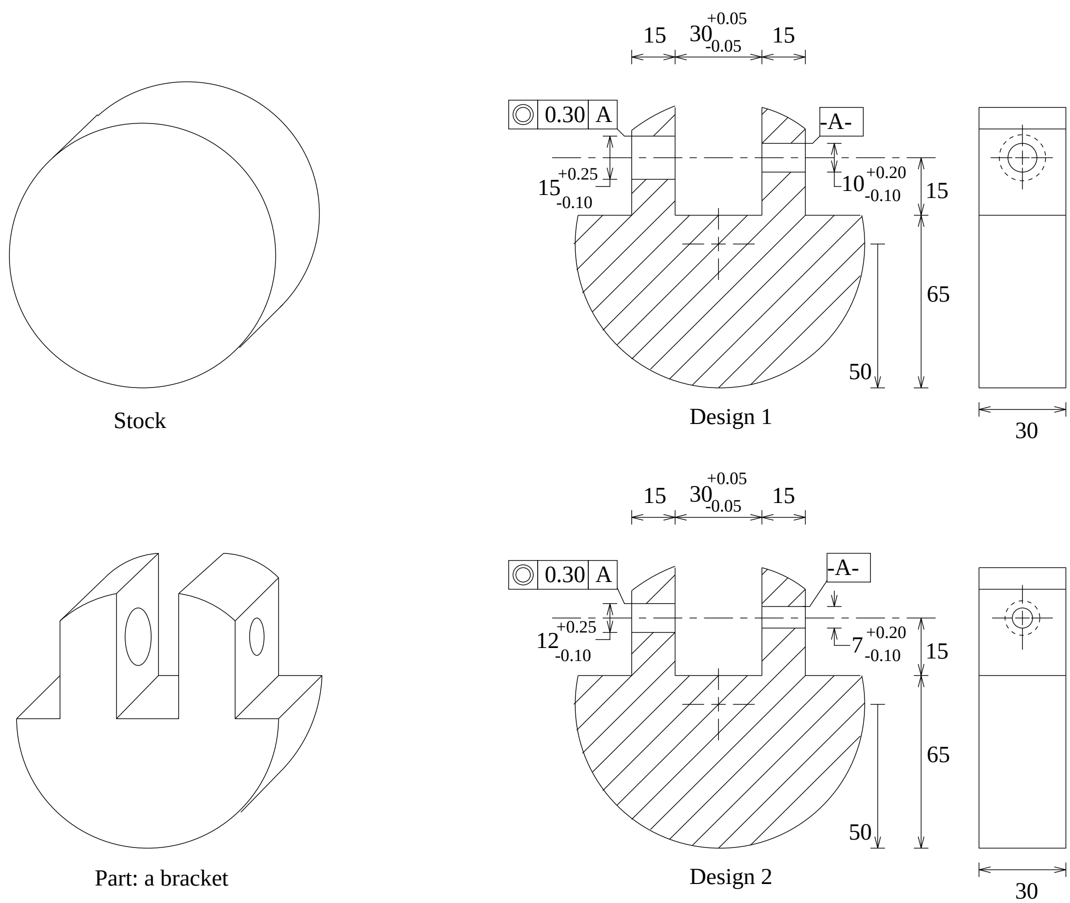


Figure 10: Two possible designs for a bracket.

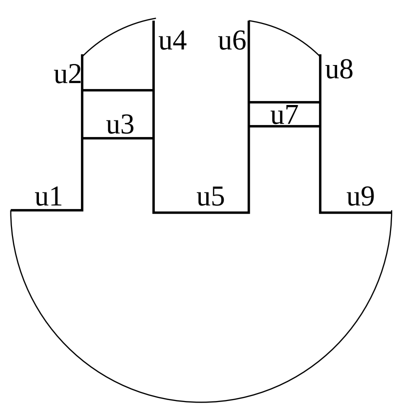


Figure 11: Faces to be created for Design 1. For Design 2, the faces are similar, except that they have different dimensions.

machined on a vertical machining center and end-milling and drilling are the only available machining operations. Fig. 11 gives names to various faces which need to be created in the two designs.

For Design 1, Fig. 12 shows the set of features $\mathcal{F}$ identified using our algorithm. From the surface $u1$, we identified features $s1$ and $s4$. For Design 2, the set of features identified by our algorithm is identical to $\mathcal{F}$, except that the holes have different dimensions.

6 Generating Feature-Based Models

Many times, the set $\mathcal{F}$ of all primary features contains redundant features. Thus, for the purpose of operation planning, we consider various subsets of $\mathcal{F}$ that are sufficient to create the part P from S and contain no redundant feature. These sets, which we call feature-based models, are defined below.

Feature-Based Models. A set of features F is *feature-based model* (or FBM) of P, S and $\mathcal{M}$ if it has the following properties:

1. For every feature f in F, its machining operation op(f) is in $\mathcal{M}$.
2. If we subtract the features in F from S, we get P (i.e., $S -^* \bigcup_{f \in F}(\text{rem}(f)) = P$).
3. No feature f in F is redundant (i.e., $S -^* \bigcup_{g \in F - \{f\}}(\text{rem}(g)) \neq P$).

Intuitively, an FBM is an interpretation of the delta volume as a set of machining features.

Primary FBMs. An FBM F is *primary* if all features in F are primary (i.e., $F \subseteq \mathcal{F}$).

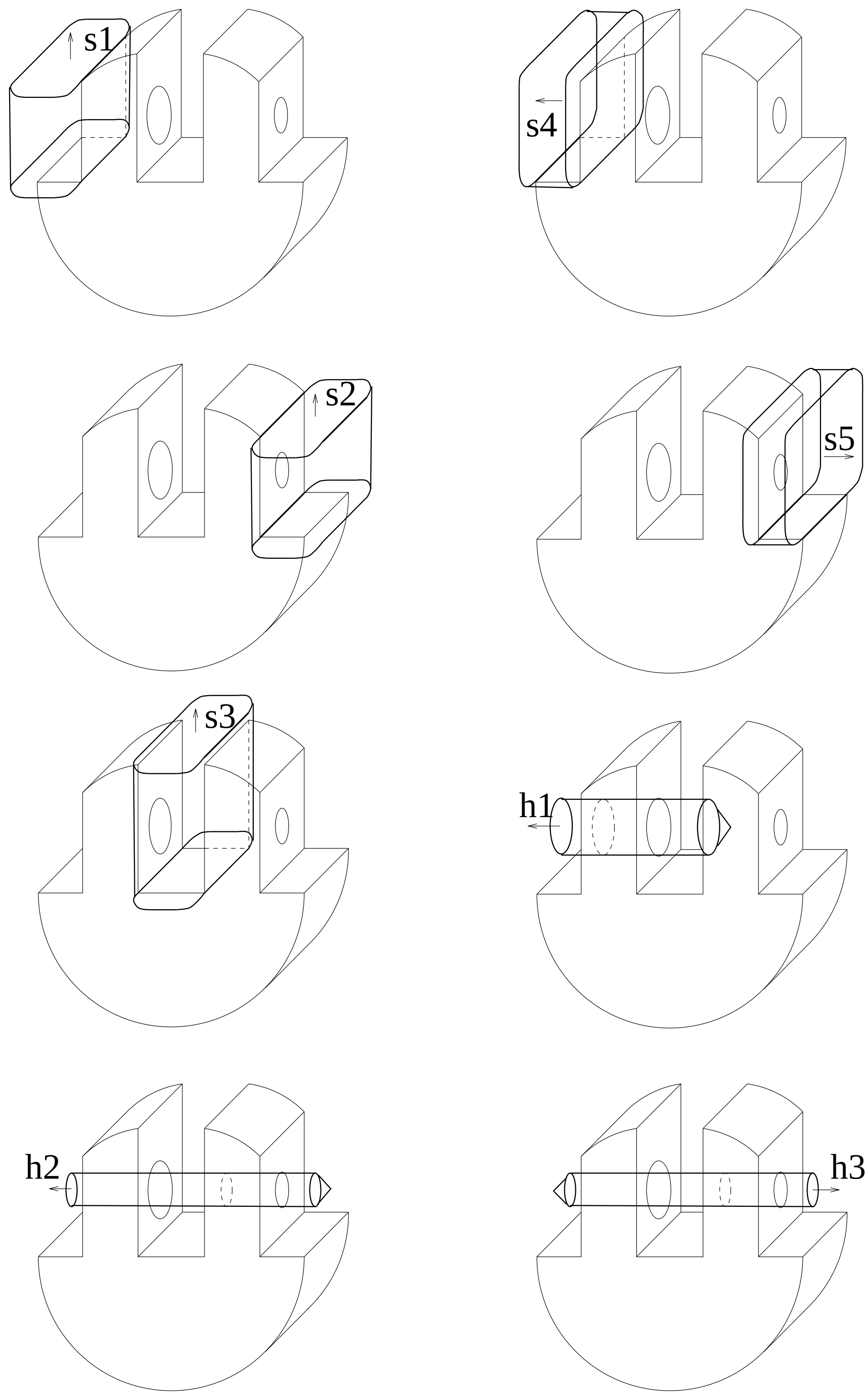
Algorithms for generating primary FBMs. Since each primary FBM is basically a set cover for the set $\mathcal{F}$, we will generate FBMs using set-covering techniques, and use pruning heuristics to discard unpromising FBMs. The algorithms for this are shown below. Basically, these algorithms operate as follows.

INITIALIZE assigns initial values to some variables, and calls GENERATE-COVERS. GENERATE-COVERS is a backtracking algorithm that looks for sets of effective removal volumes that form irredundant set-covers for the delta volume. Each such set cover corresponds to one or more FBMs. Thus, for each set cover R that GENERATE-COVERS finds, it calls COVER-TO-FBMS to find one or more FBMs F such that the effective removal volumes of the features in F are identical to the volumes in R.

procedure INITIALIZE($\mathcal{F}$)

1. For the features in $\mathcal{F}$, let $\mathcal{R}$ be the set of effective removal volumes with respect to S (i.e., $\mathcal{R} = \{\overline{\text{rem}}(f, S) : f \in \mathcal{F}\}$). For example, Fig. 13 shows $\mathcal{R}$ in the case where $\mathcal{F}$ is as shown in Fig. 12.
2. Initially, let R contain every volume in $r \in \mathcal{R}$ that is not subsumed by the other volumes in R (i.e., $R = \{r : r -^* \bigcup_{q \in \mathcal{R} - \{r\}}(q) \neq \emptyset\}$). For example, in Fig. 13, $R = \mathcal{R}$. Note that each volume $r \in R$ is guaranteed to be in every irredundant cover for $\mathcal{R}$.
3. Call GENERATE-COVERS($\mathcal{R} - R, R$).

available machining operations = $\mathcal{M}$ = {drilling, end-milling}



features identified = $\mathcal{F} = \{s1, s2, s3, s4, s5, h1, h2, h3\}$

Figure 12: Features identified by our algorithm for Design 1 of Fig. 10. For Design 2, the features are identical, except that the holes have different diameters.

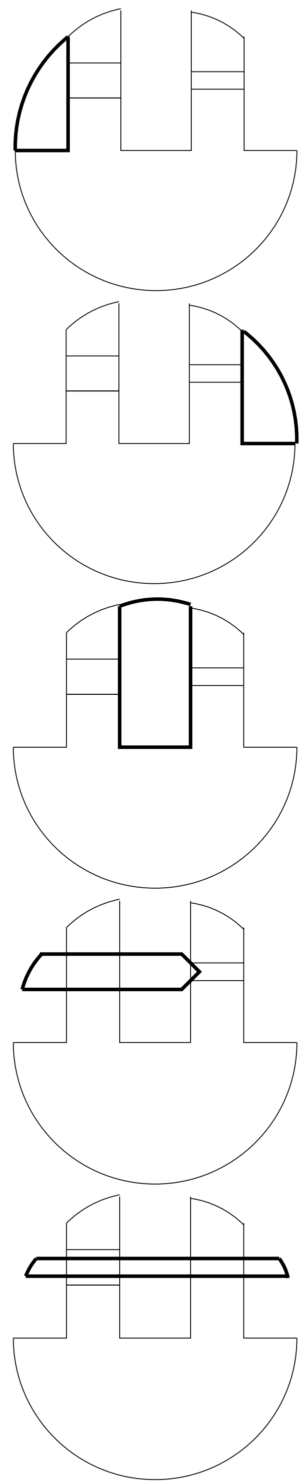
$$r1 = \overline{\mathrm{rem}}(s1, S) = \overline{\mathrm{rem}}(s4, S)$$

$$r2 = \overline{\mathrm{rem}}(s2, S) = \overline{\mathrm{rem}}(s5, S)$$

$$r3 = \overline{\mathrm{rem}}(s3, S)$$

$$r4 = \overline{\mathrm{rem}}(h1, S)$$

$$r5 = \overline{\mathrm{rem}}(h2, S) = \overline{\mathrm{rem}}(h3, S)$$



$$\mathcal{R} = \{r1, r2, r3, r4, r5\}$$

Figure 13: Effective removal volumes of features in $\mathcal{F}$, where $\mathcal{F}$ is as shown in Fig. 12.

Below, GENERATE-COVERS takes two arguments, X and R. R is the partial set cover that has been built up already, and X is a set of volumes that can potentially be added to R to complete the set cover. GENERATE-COVERS calls itself recursively to remove elements from X and add them to R, calling COVER-TO-FBMS each time this results in a complete cover. For example, in Fig. 13, INITIALIZE calls GENERATE-COVERS with $X = \emptyset$ and $R = \{r1, r2, r3, r4, r5\}$. In this case, R is already an irredundant cover (in fact, the only possible irredundant cover) so GENERATE-COVERS calls COVER-TO-FBMS immediately.

In cases where X is nonempty, the efficiency of GENERATE-COVERS depends on the order in which it chooses the volumes in X. To make the procedure efficient, in Step 4(a) our heuristic is to choose the volume r in X which covers the maximum portion of the uncovered delta volume (i.e., choose a $r \in X$ such that $r \cap^* (\Delta -^* \cup(R))$ is maximized).

procedure GENERATE-COVERS(X, R)

1. If R contains a volume r that is subsumed by the other volumes in R (i.e., $\cup^*(R - \{r\}) = \cup^*(R)$), then return, because R is redundant.

2. Otherwise, if the delta volume is completely covered by R (i.e., $\Delta \subseteq \cup^*(R)$), then call COVER-TO-FBMS($R, \emptyset$) and return, because we have found an irredundant cover.

3. Otherwise, if the volumes in R and X cannot cover the delta volume (i.e., $\Delta \supset \cup^*(R \cup X)$), then return, because R is not feasible.

4. Otherwise, do the following:

   (a) Choose a volume r in X (see discussion above).

   (b) Call GENERATE-COVERS($X - \{r\}, R \cup \{r\}$).

   (c) Call GENERATE-COVERS($X - \{r\}, R$).

Each time that GENERATE-COVERS finds an irredundant cover for the delta volume, the next step is to generate one or more primary FBMs from this cover. We do this using the depth-first branch-and-bound algorithm COVER-TO-FBM described below. COVER-TO-FBMS takes two arguments, R and G. G is the partial FBM that has been built up already, and R is the set of volumes from which features need to be generated in order to complete G. COVER-TO-FBMS calls itself recursively to remove volumes from R, and try alternative completions of G consisting of alternative features corresponding to these volumes. For each primary FBM that COVER-TO-FBM generates, it calls the algorithm GENERATE-PLANS (described in the next section), in order to map the FBM into one or more operation plans and evaluate their manufacturability.

If good FBMs have been generated and examined first, then we need not examine any FBM that is not expected to result in a better operation plan. Thus, the computational efficiency of COVER-TO-FBMS is dependent on the order in which COVER-TO-FBMS removes volumes from R and adds features to G. To control the order in which FBMs are generated, we use heuristic techniques, as described in the algorithm.

procedure COVER-TO-FBMS(R, G)

1. If $h(G, R)$ is worse than the manufacturability rating of the best operation plan seen so far, then return, because G is unpromising.
   (The pruning heuristic $h(G, R)$ estimates the highest possible manufacturability rating for any operation plan resulting from features in set G. This heuristic is described below.)

2. Otherwise, if $R = \emptyset$, then call GENERATE-PLANS(G) and return, because we have found a promising FBM.

3. Otherwise, do the following:

   (a) Choose an effective removal volume r in R.[1]

   (b) Since more than one feature in $\mathcal{F}$ can have r as its effective removal volume, let Feat be the set of all such features (i.e., Feat $= \{f : \overline{\text{rem}}(f, S) = r\}$).

   (c) For each feature $g \in$ Feat,[2] call COVER-TO-FBMS($R - \{r\}, G \cup \{g\}$).

Pruning Heuristic $h(G, R)$. For the purposes of this paper, the manufacturability rating for a given operation plan will be the sum of setup and machining times (see Section 9 for details). Therefore, we define the heuristic function $h(G, R)$ to give the lower bound on the required setup and machining time for any operation plan resulting from features in G.

Each time that COVER-TO-FBMS is called, R is a set of effective removal volumes, and G is a set of features such that $R \cup \{\overline{\text{rem}}(g, S) : g \in G\}$ is an irredundant cover for the delta volume. For all sets R and G that satisfy this property, we define

$$h(G, R) = L_s(G) \times T_s + \sum_{g \in G} L_m(g),$$

where

- $L_s(G)$ is a lower bound on the number of setups needed to machine G. For three-axis machining centers, $L_s(G)$ is the cardinality of the set $\{\vec{v}(g) : g \in G\}$, where $\vec{v}(g)$ is the unit orientation vector for feature g.

- T_s is the average setup time.

- $L_m(g)$ is a lower bound on the time required to machine feature g. This is the time required to machine the irredundant portion of the effective removal volume of g (i.e., $\overline{\text{rem}}(g, S) -^* \cup(R) -^* \cup_{f \in G - \{f\}}(\overline{\text{rem}}(f, S))$.

How to calculate the average setup time and the machining time is discussed in Section 9.

Example. Suppose we call COVER-TO-FBMS with $R = \{r1, r2, r3, r4, r5\}$ and $G = \{\}$, where $r1, r2, r3, r4, r5$ are as shown in Fig. 13 (these effective removal volumes are for Design 1). As we will show later, one of the operation plans resulting from FBM 1 satisfies the design tolerances and has a relatively low machining time. After generating FBM 1, COVER-TO-FBMS will retain this plan as the best plan seen so far. In comparison with this plan, the value of the pruning heuristic will be high for FBMs 2, 3, 6, 7, and 8, because they require a larger number of setups. Thus, as illustrated in Fig. 14, our pruning heuristic will allow us to prune these FBMs, retaining only FBMs 1 and 5. FBMs 1 and 5 are shown in Fig. 15.

As we will show later, for Design 2, none of the FBMs results in an operation plan which satisfies design tolerances. Therefore, we need to examine all eight FBMs.

[1] The efficiency (but not the correctness) of the algorithm depends on which effective removal volume is chosen. Our heuristic is to choose the one that has minimum number of features associated with it, i.e., to choose $r \in R$ that minimizes the cardinality of the set $\{f : \overline{\text{rem}}(f, S) = r\}$.

[2] The efficiency of the algorithm depends on the order in which it examines the features in Feat. Our heuristic is examine features $g \in$ Feat in order of increasing value of the pruning heuristic $h(G \cup \{g\}, R - \{r\})$.

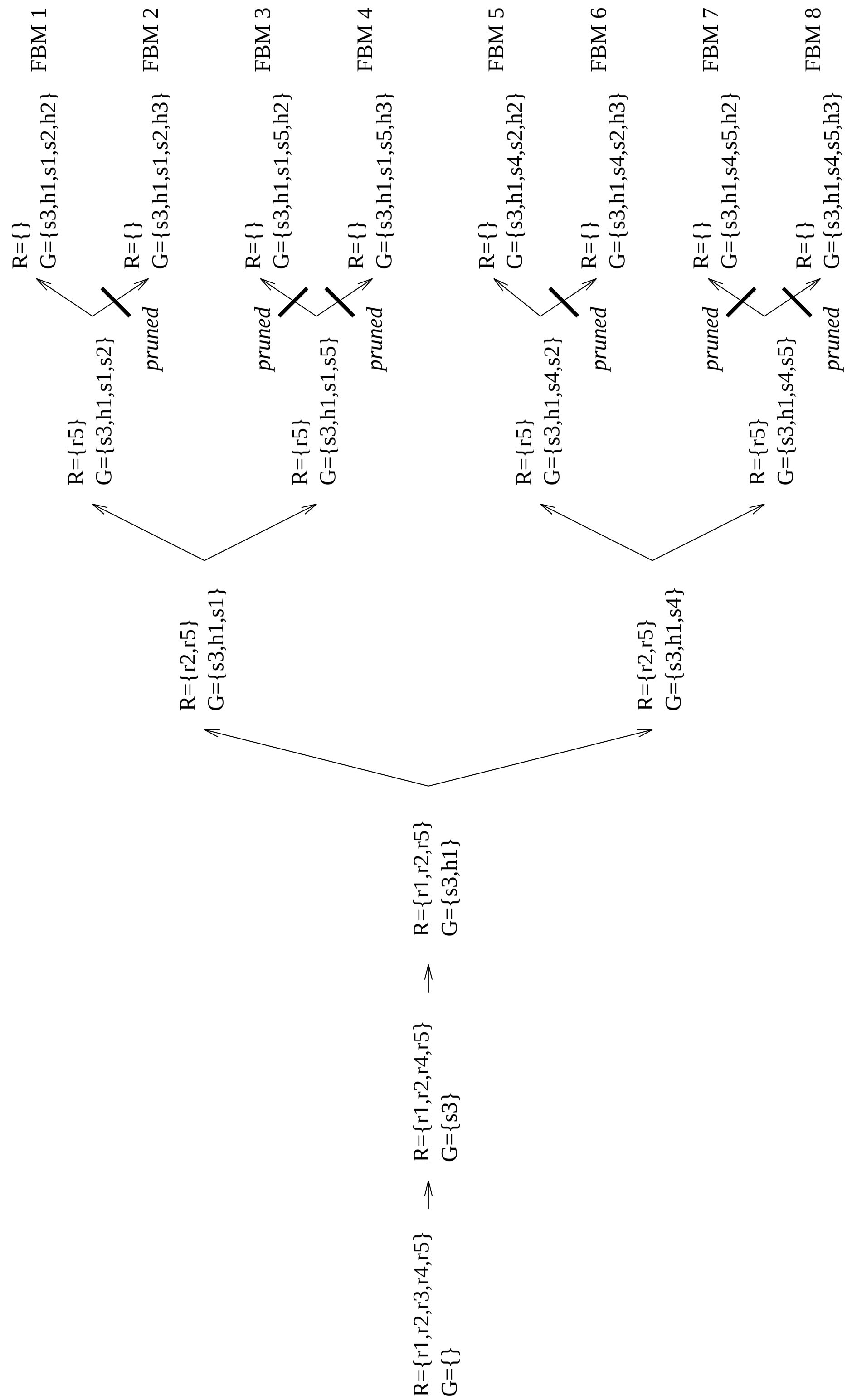
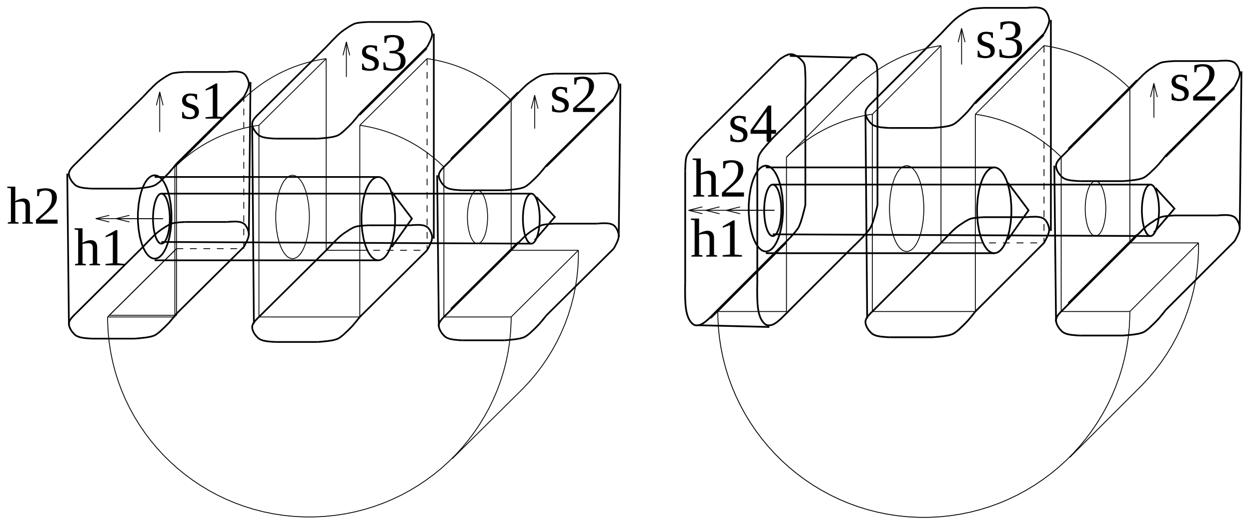


Figure 14: Recursion tree generated by the call COVER-TO-FBMs $(\{r1, r2, r3, r4, r5\}, \{\})$. In each case, R is the set of effective removal volumes and G is the set of features.



FBM 1: $\{s3, h1, s1, s2, h2\}$ FBM 5: $\{s3, h1, s4, s2, h2\}$

Figure 15: The two primary FBMs generated by our algorithm.

7 Generating Operation Plans

Due to various types of interactions (accessibility, setup, and so forth) among the features in an FBM F, the features of F cannot be machined in any arbitrary order. Instead, these interactions introduce precedence constraints requiring that some features of F be machined before or after other features. We will be interested in determining these precedence constraints because our estimates of cost and machining accuracy will depend on them.

Ordered FBMs. An *ordered FBM* (F, C) consists of an FBM F along with a set of precedence constraints C, such that the following conditions are satisfied:

1. *Accessibility.* C contains a constraint $f \to f'$ for every pair of operations f, f' such that f' will not be accessible until we machine f.

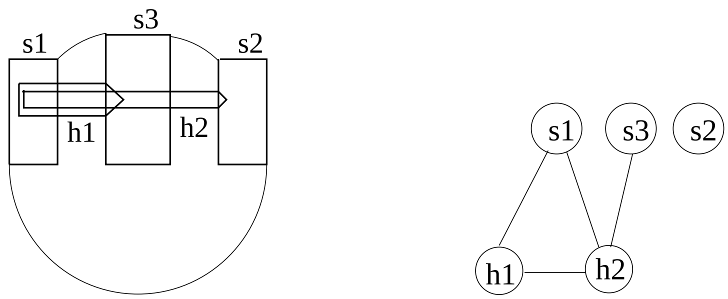
2. *Feature minimality.* C contains a constraint $f \to f'$ for every pair of operations f, f' such that machining f' before f would allow us to machine the volume machined by f using a smaller feature.

Operation Plans. An *operation plan* (O, C) is a set of machining operations O (along with the recommended cutting parameters) and a set of precedence constraints C on the order in which the operations are to be performed.

Normally, we will only be interested in operation plans that can be generated from ordered FBMs. As we will discuss in more detail later, generating an operation plan (O, C) from an ordered FBM (F, C) involves mapping each feature f of F to the machining operation $\text{op}(f)$ capable of creating the feature, and applying[3] the precedence constraints C to the machining operations.

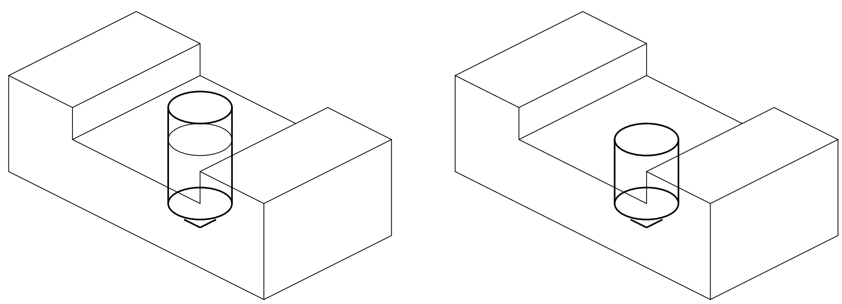
Algorithm for generating operation plans. After generating primary FBMs, the next step is to generate the associated machining operations along with their precedence constraints. We generate operation plans as follows:

[3] Mathematically, this means extending C by defining a constraint $\text{op}(f) \to \text{op}(f')$ for every constraint $f \to f'$ in C.



(a): FBM 1 after trimming (b): each edge represents a significant intersection between two features

Figure 16: Significant intersections among features in FBM 1. Note that the the intersection between a drilling feature's conical bottom and any other feature is not significant.



(a): before truncation (b): after truncation

Figure 17: An example of feature truncation.

procedure Generate-Plans(F)

1. Call Trim-Primary-FBM(F). This procedure, defined below, trims the unuseful portions of the feature in FBM F.

2. Let $\{(f_1, f_1'), (f_2, f_2'), \ldots, (f_n, f_n')\}$ be all pairs of features in F that have significant intersections.[4] For every consistent set C of precedence constraints on these pairs of intersecting features, i.e., every consistent set C of precedence constraints such that for each i, C contains either the constraint $f_i \to f_i'$ or the constraint $f_i' \to f_i$ but not both,[5] do the following:

   (a) Let $f_1, f_2, \ldots, f_n$ be any total ordering of F that is consistent with C. (Such a total ordering can easily be generated using topological sorting [6]. This total ordering is not unique, but since C totally orders intersecting features, we can prove that we will get exactly the same operation plan regardless of which total ordering is produced by the topological sorting algorithm.)

   (b) For every $i > 0$, let g_i be the *truncation* of f_i with respect to the workpiece $W_i = S -^* (f_1 \cup^* \ldots \cup^* f_{i-1})$. By this, we mean the smallest feature g of f's type and orientation such that g can remove the volume removed by f from W_i, i.e., $\overline{\mathrm{rem}}(g, W_i) = \overline{\mathrm{rem}}(f, W_i)$ (for

[4] Two features f_1 and f_2 have a *significant* intersection if the intersection is expected to result in either accessibility or minimality type of precedence constraint after truncating the features in Step 2b. For an example, see Fig. 16.

[5] In the worst case, this could be a very large number of sets of precedence constraints—but we believe that this worst case is quite unlikely to occur, because in cases, all sets of significantly intersecting features will be quite small.

example, see Fig. 17). If f_i violates any dimensional constraints, or is not accessible in W_i after truncation, then discard C and skip Step (c).

(c) Let $G = \{g_1, \ldots, g_n\}$. Add the ordered FBM (G, C) to the set $\mathcal{G}$ of all ordered FBMs found so far.

3. Call Relax-Prec-Const($\mathcal{G}$). This procedure, defined below, detects situations in which some of the precedence constraints can be eliminated.

4. For every ordered FBM $(G, C) \in \mathcal{G}$, do:

   (a) Generate the operation plan (O, C) by mapping each g in G to its associated machining operation o (o consists of op(g) plus the recommended machining parameters taken from a machining data handbook such as [18]), and applying the precedence constraints C to the machining operations.

   (b) Call Evaluate-Mach-Accr(O, C). This procedure, defined in Section 8, estimates the achievable tolerances for the operation plan (O, C).

   (c) If (O, C) is capable of meeting the design tolerances then compute the manufacturability rating M_R for the operation plan (O, C), as described in Section 9.

Trimming of unuseful portions of features in primary FBMs. In a primary FBM, the removal volumes of various features may sometimes intersect. If two features have intersecting removal volumes, then machining either of them will remove the shared volume. In some cases, it may be possible to modify one of the features by modifying it to remove some or all of the shared volume, so as to avoid machining it twice. Trimming only eliminates the bottom portion of the feature without affecting its datum point or approach face (if possible, remaining portions will get truncated in Step 2b of Generate-Plan procedure). However, since it changes the size of the feature, the resulting feature (and thus the FBM) will not be primary. In order to produce more realistic estimates of cost and machining accuracy, the first step in Generate-Plans(F) is to trim the features in the FBM F wherever it can. This is done as described below.

procedure Trim-Primary-FBM(F)

1. For every $f \in F$, do the following:

   (a) Let r be the portion of f's removal volume that is not shared by any other feature, i.e., $r = (\overline{\text{rem}}(f, S) -^* \cup_{g \in F - \{f\}}(\text{rem}(g))$.

   (b) Find the smallest feature h that satisfies following conditions:

      i. h and f have the same machining operations (i.e., op(h) = op(f)).

      ii. h and f have the same orientation vector (i.e., $\vec{v}(h) = \vec{v}(f)$).

      iii. h and f have the same datum points (i.e., $p_d(h) = p_d(f)$).

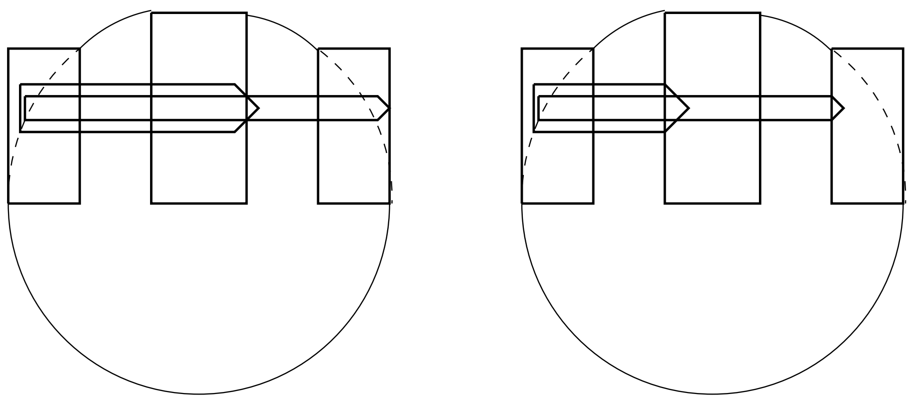
      iv. h and f have the same approach face (i.e., $a(h) = a(f)$).

      v. h can remove the volume r (i.e., $\overline{\text{rem}}(h, S) \supseteq r$).

   (c) Add h to H.

2. If the features in H can create P from S (i.e., $S -^* \bigcup_{h \in H} \text{rem}(h) = P$), then return H. Otherwise, return F.

Fig. 18 shows the result of applying the above procedure to FBM 1 of Fig. 15.



(a): before trimming (b): after trimming

Figure 18: Trimming FBM 1 (of Fig. 15).

Relaxing Precedence Constraints. Step 2 of the procedure Generate-Plans may generate FBMs with identical features but different precedence constraints. In such cases, it is possible to combine the sets of precedence constraints into a single set by relaxing the conflicting precedence constraints.

For example, suppose that after Step 2, we get two ordered FBMs (G, C) and (G, C') containing the same FBM G, such that C contains a precedence constraint $g_1 \rightarrow g_2$, and C' contains the precedence constraint $g_2 \rightarrow g_1$. Then both of these precedence constraints are unneeded, and can be removed from C and C'.

It may not be immediately obvious to the reader why the above property holds, but the explanation is as follows. The precedence constraint $g_1 \rightarrow g_2$ is not needed for ensuring accessibility in the operation plan resulting from (G, C), because g_2 was accessible before g_1 in (G, C'). The precedence constraint $g_1 \rightarrow g_2$ is not needed for ensuring feature minimality in (G, C), because (G, C') was generated by truncating g_2 before g_1, and it did not result in any smaller feature than g_2. Therefore, the precedence constraint $g_1 \rightarrow g_2$ is not needed in (G, C). Similarly, the precedence constraint $g_2 \rightarrow g_1$ is not needed in (G, C).

Step 3 of Generate-Plans takes advantage of the above property to relax the precedence constraints by discarding conflicting constraints. This is done using the following procedure:

procedure Relax-Prec-Const($\mathcal{G}$)

1. While $\mathcal{G}$ contains ordered FBMs (G, C') and (G, C'') containing the same FBM G but different precedence constraints (i.e., $C' \neq C''$), do the following:
   (a) Let C be the set of precedence constraints consisting those precedence constraints that are common in C' and C'' (i.e., $C = C' \cap C''$).
   (b) Replace (G, C') and (G, C'') in $\mathcal{G}$ by (G, C).
2. Return $\mathcal{G}$.

Examples. Fig. 19 shows the results produced by the first three steps of Generate-Plan on FBM 1. As shown in the figure, Step 2 produces four FBMs, of which the two leftmost ones are identical. Therefore, in Step 3, Generate-Plan relaxes the precedence constraints $s3 \rightarrow h2$ and $h2 \rightarrow s3$, producing the ordered FBM shown in the lower left-hand corner of the figure.

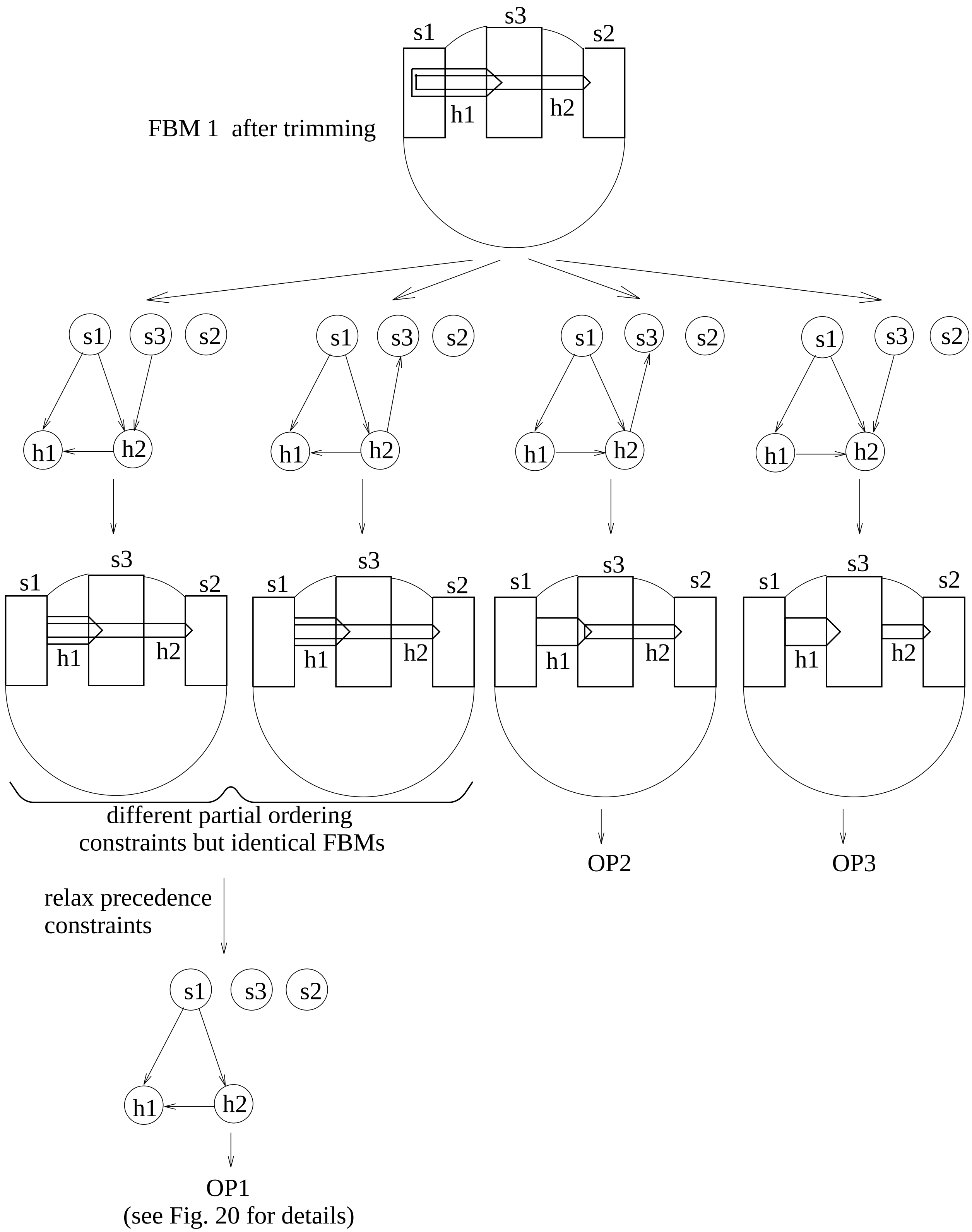


Figure 19: Generating operation plans from FBM 1.

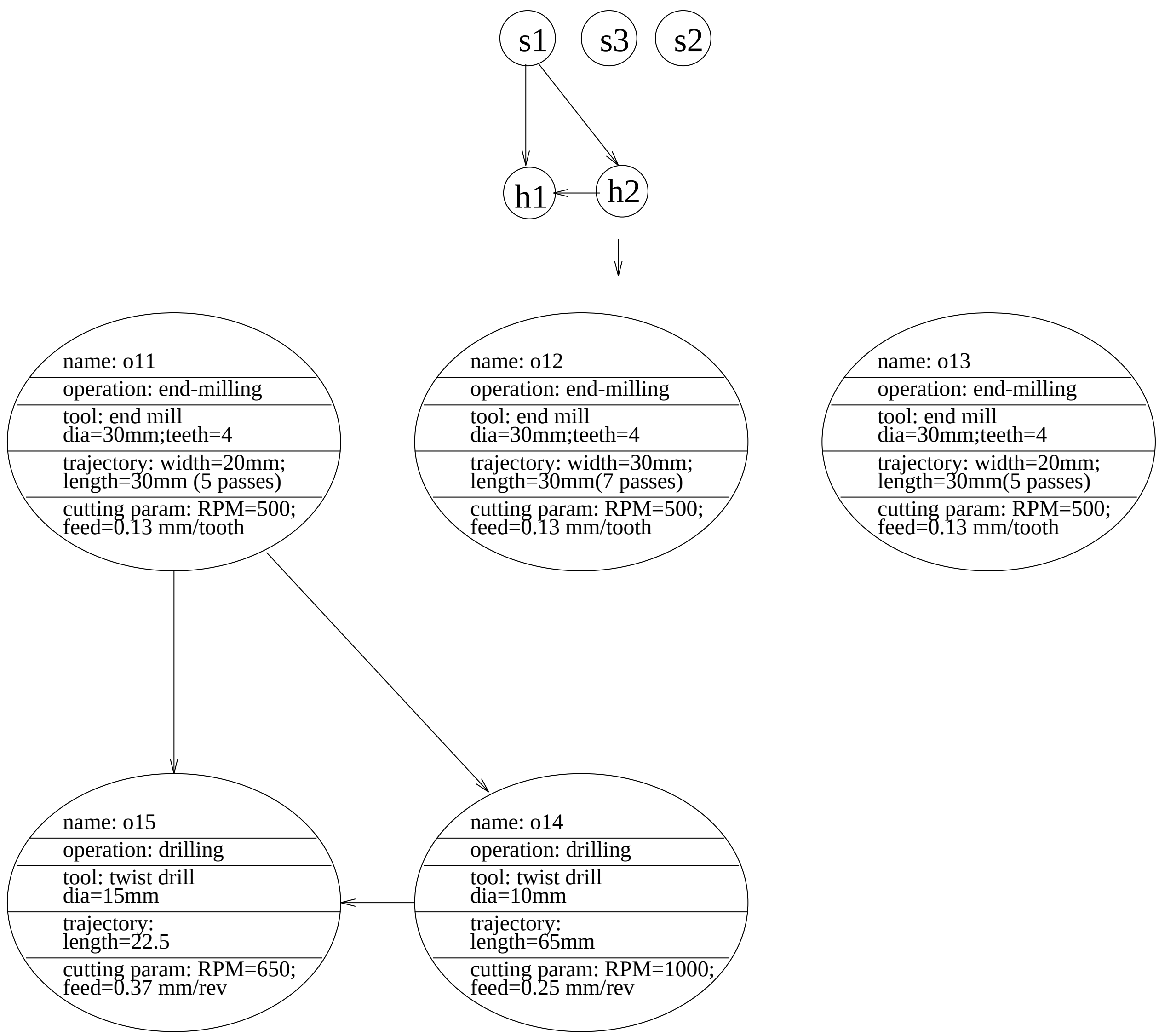
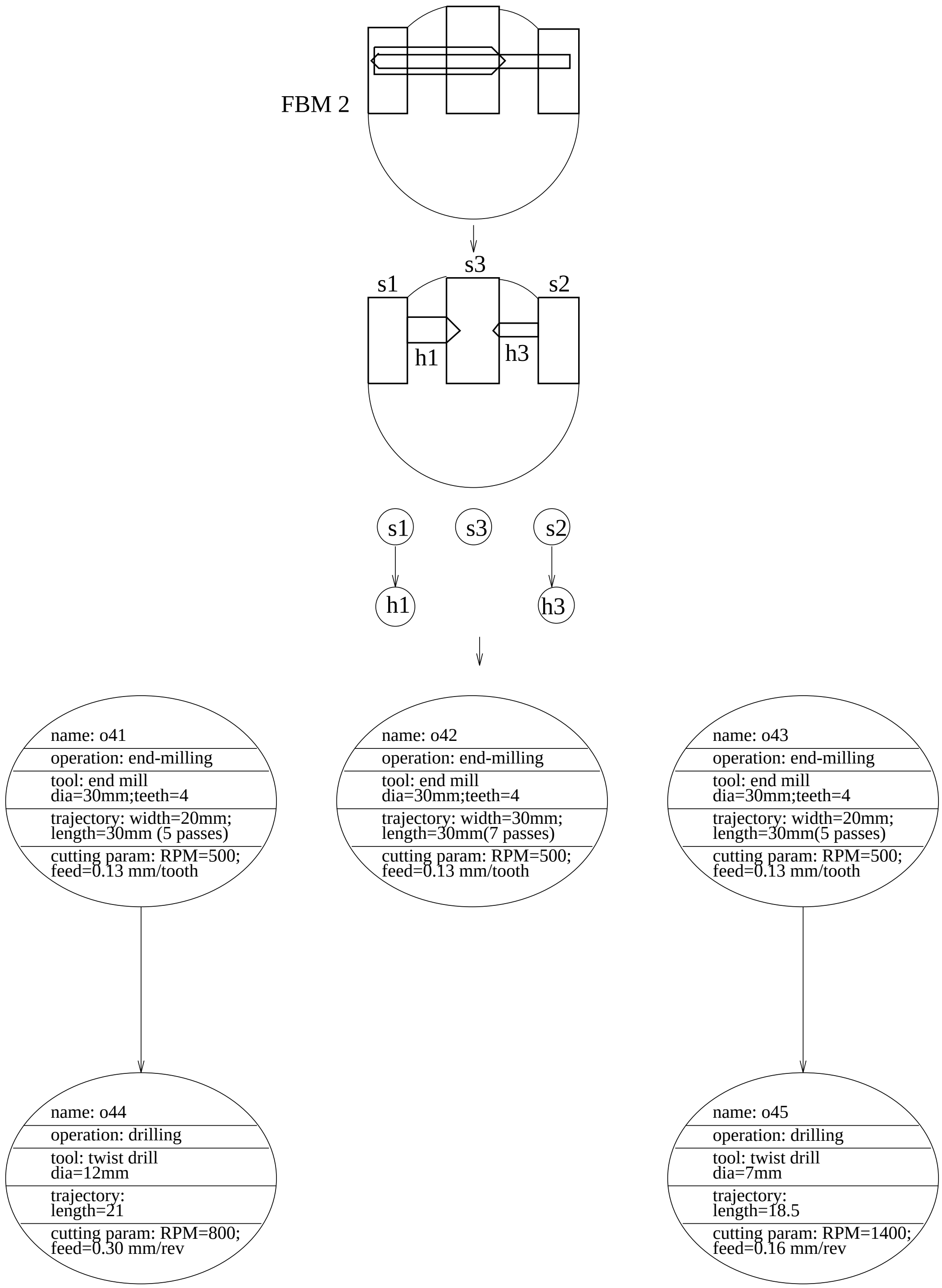


Figure 20: Details of Operation Plan OP1 for Design 1.

Because of the cylindrical stock boundary, if the hole $h1$ is machined before the slot $s1$, then h_1's entry face will be a curved surface and will pose an accessibility problem. Therefore, procedure GENERATE-PLANS (F) will generate no ordered FBM in which $h1 \rightarrow s1$. Some of the dimensional constraints used on available machining operations are given below:

- For drilling tools, the maximum *length/diameter* ratio is 8.
- For end-milling operations, the maximum *depth of cut* is 6mm.
- For end-milling tools, the maximum *tool diameter* is 30mm.

Fig. 20 gives details of operation plan OP1 for Design 1. As described in Sections 8 and 9, this operation plan is capable of producing the desired tolerances for Design 1 and requires the minimum production time.



Details of Operation Plan OP4 for Design 2.

Figure 21: Generating operation plan from FBM 2.

Table 2: Machining accuracy data for cylindrical surfaces created by drilling operations.

diameter range (mm)	dimensional variation (mm)	form error (mm)	surface finish (μm)
3 to 6	+0.100, −0.025	0.100	1.6
6 to 13	+0.150, −0.025		
13 to 25	+0.200, −0.050		
25 to 50	+0.250, −0.080		
50 to 100	+0.300, −0.100		

Table 3: Machining accuracy data for planar surfaces created by end-milling operations.

dimensional variation (mm)	form error (mm)	surface finish (μm)
+0.020, −0.020	0.040	0.8

If we start from Design 2 rather than Design 1, we will get FBMs identical to FBMs 1 through 8, except for the dimensions of the holes. However, because of the restriction on the maximum length/diameter ratio of drilling tools, Design 2 can only be machined by drilling the two holes from opposite approach directions—and thus no operation plans will be generated from FBM 1, FBM 3, FBM 5 or FBM 7. Fig. 21 shows generation of operation plans from FBM 2 for Design 2. As described in Section 8, no operation plan is capable of producing the desired tolerances for Design 2.

8 Estimating Achievable Machining Accuracy

Each machining operation creates surfaces that have certain geometric variations compared to its nominal geometry. Designers normally give *design tolerance* specifications on the nominal geometry, to specify how large these variations are allowed to be without violating the functionality requirements. To verify whether or not a given operation plan will produce the desired design tolerances, we want to estimate what tolerances the operations can achieve.

Currently, we handle tolerances only on planar and cylindrical surfaces. Each machined surface has following three associated accuracy characteristics:

1. *A form error*, ϵ_f. For planar surfaces this error is the flatness of the machined surface, and for cylindrical surfaces is the cylindricity of the machined surface.

2. *A dimensional variation*, ϵ_d. This defines the dimensional zone with respect to the nominal (ideal) surface in which the machined surface lies. For planar surfaces, this zone is defined by two planar surfaces parallel to the nominal surface, and for cylindrical surfaces, it is defined by two cylindrical surfaces concentric with the nominal surface. The two components of ϵ_d are represented by ϵ_d^+ and ϵ_d^-.

3. *A surface finish*, ϵ_s. This describes the smoothness (or roughness) of the machined surface.

Each machining operation creates surfaces with different characteristics. Table 2 presents machining accuracy data for cylindrical surfaces produced by drilling operations, compiled from [3, 5]. Table 3 presents machining accuracy data for planar surfaces produced by end-milling operations, compiled from [3, 5].

Table 4: Location errors for drilling operations.

drilling condition	location error
drilling a new hole	0.15mm
enlarging a pre-existing hole	0.10mm

Table 5: Tolerance table for Design 1.

surface(s)	tolerance type	value(s)
$u3$	diameter	$+0.25, -0.10$
$u7$	diameter	$+0.20, -0.10$
$(u4, u6)$	length	$+0.05, -0.05$
$(u3, u7)$	concentricity	0.30

For some machining operations, the actual datum point is different from the desired datum point. This error is refered as the location error, ϵ_l. For example, in case of drilling operations, the center of a drilled hole is different from the nominal (ideal) location. Table 4 gives the data for location errors in drilling operation, compiled from [3, 5]. We assume this error to be zero for end-milling operations.

From the design tolerances, we construct a tolerance table which describes these tolerances as attributes of various surfaces. For example, Table 5 is the tolerance table for Design 1. Since Design 2 has tolerances identical to Design 1, the tolerance table for Design 2 is same as Table 5. A tolerance table has two types of entries: tolerances associated with a single surface, and tolerances associated with a pair of surfaces.

We use the following algorithm to evaluate the machining accuracy of a given operation plan:

procedure EVALUATE-MACH-ACCR(O, C)

1. Let $\{o_1, o_2, \ldots, o_n\}$ be a total order of the operations in O that is consistent with precedence constraints in C.

2. For $i > 0$, if o_i creates any face u which appears in the tolerance table, then calculate the form error ϵ_f, dimensional variation $\epsilon_d^+, \epsilon_d^-$, surface finish ϵ_s, and location error ϵ_l associated with u. Let $E(u) = \{\epsilon_f, \epsilon_d^+, \epsilon_d^-, \epsilon_s, \epsilon_l\}$.

3. For every tolerance-table entry that involves a single surface u, do the following:

   (a) Calculate the achievable tolerance from $E(u)$ (some of the formulas we use for this are described below).

   (b) If the achievable tolerance does not satisfy the design tolerance, then return failure.

4. For every tolerance-table entry that involves a pair of surfaces u and u', do the following:

   (a) Calculate the achievable tolerance from $E(u)$ and $E(u')$, using different setups for u and u'.

   (b) If the achievable tolerance does not satisfy the design tolerance, then add (u, u') to the set $\mathcal{U}$. $\mathcal{U}$ contains pairs of features whose tolerance is tight enough that it cannot be achieved in different setups.

5. If there is no way to order the operations in O consistently with C such that the operations associated with every pair $(u, u') \in \mathcal{U}$ can be done in a single setup, then return failure.

6. For every $(u, u') \in \mathcal{U}$, do the following:
   (a) Calculate the achievable tolerance from $E(u)$ and $E(u')$ assuming the same setup for u and u'.
   (b) If the achievable tolerance does not satisfy the the design tolerance, then return failure.
7. Return success. (Note that this will occur only if Steps 3(b), 5 or 6(b) do not return failure.)

Below are examples of the formulas used to calculate the tolerances in above algorithm (the others are omitted for brevity):

1. *Length Tolerance.* Let $u1$ and $u2$ be two parallel planar surfaces. Let $(\epsilon_{d1}^{+}, \epsilon_{d1}^{-})$ and $(\epsilon_{d2}^{+}, \epsilon_{d2}^{-})$ be the dimensional variation associated $u1$ and $u2$ respectively. Then the length tolerance between $u1$ and $u2$ is defined as
$$\begin{aligned} \text{upper limit} &= \epsilon_{d1}^{+} + \epsilon_{d2}^{+} + \text{setup error} \\ \text{lower limit} &= \epsilon_{d1}^{-} + \epsilon_{d2}^{-} + \text{setup error} \end{aligned}$$
2. *Concentricity Tolerance.* Let $u1$ and $u1$ be two cylindrical surfaces. Let ϵ_{l1} and ϵ_{l2} be the location error associated with the datum points of $u1$ and $u2$ respectively. Then the concentricity tolerance between $u1$ and $u2$ is defined as
$$\text{concentricity tolerance} = \epsilon_{l1} + \epsilon_{l2} + \text{setup error}$$

The setup error is assumed to be 0 if the operations associated with $u1$ and $u2$ are done in same setup, and 0.20mm otherwise.

Examples. Operation plan OP1 for Design 1 is capable of producing the desired tolerances. All plans for Design 2 that satisfy length/dia restriction for drilling tools (for example, plan OP4 shown in Fig. 21) require a setup change between two drilling operations, and thus none of these plans can produce the desired concentricity tolerance for Design 2. Therefore, Design 2 cannot be machined using drilling and end-milling operations.

9 Manufacturability Rating of Designs

The manufacturability of a given design depends on the following three factors:

1. the ability to produce the design within the specified specification;
2. the ability to produce the design with a low production cost;
3. the ability to produce the design with a low production time.

The first of these factors was handled in the previous sections: if the design cannot be produced within the desired specifications, then our approach returns failure. If the design can be produced within the desired specifications, then the manufacturability rating could be based on approximations of the production cost, production time, or a combination of the two.

In this paper, we ignore the production cost, and simply use an approximation of the production time (minimized over all plans capable of producing the design) as the design's manufacturability rating. More specifically, we calculate the manufacturability rating of a design as

$$M_R(O, C) = \min\{PT(O, C) : (O, C) \text{ is an operation plan that meets the design specs}\},$$

Table 6: Time estimates for various operations in operation plan OP1

Operation	$o11$	$o12$	$o13$	$o14$	$o15$	2 setups	Total time
Time (min)	0.87	1.22	0.87	0.26	0.10	3.00	6.32

where

$$\begin{aligned} PT(O,C) &= n \times T_s + \sum_{o \in O} T_m(o); \\ O &= \text{the set of machining operations;} \\ C &= \text{the set of precedence constraints on } O; \\ T_s &= \text{the average setup time (this can be estimated} \\ &\quad\ \text{using information from handbooks such as [26]);} \\ n &= \text{the minimum possible number of setups (this can be} \\ &\quad\ \text{computed using branch-and-bound techniques [7]);} \\ T_m(o) &= \text{the machining time associated with } o \text{ (this can be estimated} \\ &\quad\ \text{using information from handbooks such as [18, 26, 27]).} \end{aligned}$$

Example. Fig. 20 shows the details of the plans OP1. This plan is estimated to require 2. For Design 1, OP1 has the best estimated production time; its estimated production time is shown in Table 6. In estimating the production time, we have added half the tool diameter to each slot length to account for the lead-in. We assumed that the part will machined on a three axis vertical machining center and will be held in a vise.

10 Conclusions and Future Work

In this paper, we have presented a methodology for analyzing proposed product designs to estimate their manufacturability aspects. Our analysis provides feedback to the designer about possible manufacturability problems, thereby providing an opportunity to redesign the product to improve its manufacturability characteristics. Some of the benefits of our approach are listed below:

1. By using features that correspond directly to machining operations, we are incorporating process-related information in the features themselves. This will allow us to estimate production cost and quality without going through a very elaborate process-planning step.

2. As opposed to existing rule-based approaches, our approach is based on theoretical foundations. We anticipate that this will enable us to make rigorous statements about the soundness, completeness, efficiency, and robustness of the approach.

3. Since we consider various alternative ways of machining the part, the conclusions about the manufacturability of the proposed design will be more realistic than if we considered just one alternative.

In this work, we have made an attempt in direction of building a systematic approach to manufacturability analysis, but the overall problem is far from solved. For example, in our current work we do not consider fixturing, which can have significant affects on operation planning. In our future work, we plan to integrate a fixturability analysis system with the current system.

Many real-life parts are produced using a combination of primary and secondary processes. For example, engine blocks are first cast, and then machined to final shape. In many cases, manufacturability requirements for different processes are very different. For example, a design shape that is easy to cast may pose problem during fixturing. Incorporating more then one type of manufacturing processes in the same manufacturability analysis system is a challenging research problem. As an extension of our current work, we would like to develop a manufacturability analysis system that will handle combinations of manufacturing processes.

We also intend to extend our approach so that in addition to identifying design problems and computing the manufacturability rating, it will also suggest changes that can improve the manufacturability of the design. As a first step in this direction, we are currently developing ways to identify changes that will reduce the number of setups.

The ultimate goal of our approach is to provide tools for manufacturability analysis as part of the CAD systems used by designers. Even if designers have access to such tools, the ultimate cost and quality of the product will still depend on the designer's creativity and ability. However, we believe that our research will help in improving the productivity of designers, by helping them to design products that are easier to manufacture. This will reduce the need for redesign, resulting in reduced lead time and product cost.

In addition, we anticipate that the results of this work will be useful in providing a way to speed up the evaluation of new product designs in order to decide how or whether to manufacture them. Such a capability will be especially useful in flexible manufacturing systems, which need to respond quickly to changing demands and opportunities in the marketplace.

Acknowledgements

This work was supported in part by NSF Grants NSFD CDR-88003012, IRI-9306580, and DDM-9201779.

References

[1] L. Alting and H. Zhang. Computer aided process planning: The state of the art survey. *Int. J. of Prod. Res*, 27(4):553–585, 1989.

[2] R. Bakerjian, editor. *Design for Manufacturability*, volume 6 of *Tool and Manufacturing Enginners Handbook*. Society of Manufacturing Engineers, 1992.

[3] James G. Bralla, editor. *Handbook of Product Design for Manufacturing*. McGraw Hill Book Company, 1986.

[4] S.L. Brooks, K.E. Hummel, and M.L. Wolf. XCUT: a rule-based expert system for the automated process planning of machined parts. In *ASME Symposium on Integrated and Intelligent Manufacturing*, Boston, MA, December 1987.

[5] Tien-Chien Chang. *Expert Process Planning for Manufacturing*. Addison-Wesley Publishing Co., 1990.

[6] T. H. Corman, C. E. Leiserson, and R. L. Rivest. *Introduction to Algorithms*. MIT Presss/McGraw Hill, 1990.

[7] D. Das, S.K. Gupta, and D.S. Nau. Reducing setup cost by automated generation of redesign suggestions. In *Submitted for publication in ASME Computers in Engineering Conference*, 1994.

[8] H.A. ELMaraghy, K.F. Zhang, and H. Chu. A function-orinted modeler prototype. In *ASME National Design Engineering Conference*, volume DE-Vol. 52, Chicago, IL, March 1993.

[9] R. Gadh, E.L. Gursoz, M.A. Hall, F.B. Prinz, and A.M. Sudhalkar. Feature abstraction in a knowledge-based critique of design. *Manufacturing Review*, 4(2):115–125, 1991.

[10] S.K. Gupta, T.R. Kramer, D.S. Nau, W.C. regli, and G. Zhang. Building MRSEV models for CAM applications. Technical Report TR 93-84, Institute for Systems Research, University of Maryland, College Park, Md-20742, 1993.

[11] S.K. Gupta, P.N. Rao, and N.K. Tewari. Development of a CAPP system for prismatic parts using feature based design concepts. *The International Journal of Advanced Manufacturing Technology*, 7:306–313, 1992.

[12] G. Harhalakis, A. Kinsey, I. Minis, and H. Rathbun. Manufacturability evaluation of electronic products using group technology. In *NSF design and manufacturing systems grantees conference*, Charlotte, North Carolina, 1993.

[13] C.C. Jara-Almonte and S. Krishnamorrthi. Evaluation of the manufacturability of machined components using axiomatic design principles. In *NSF design and manufacturing systems grantees conference*, Austin, TX, 1991.

[14] Ping Ji. A tree approach for tolerance charting. *International Journal of Production Research*, 31(5):1023–1033, 1993.

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

[16] R. R. Karinthi and D. S. Nau. An algebraic approach to feature interactions. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 14(4):469–484, April 1992.

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

[18] Machinability Data Center. *Machining Data Handbook*. Metcut Research Associates, Cincinnati, Ohio, third edition, 1980.

[19] M. Mantyla, J. Opas, and J. Puhakka. Generative process planning of prismatic parts by feature relaxation. In *ASME Advances in Design Automation*, volume DE-Vol. 19-1, 1989.

[20] R. O. Mittal, S. A. Irani, and E. A. Lehtihet. Tolerance control in the machining of discrete components. *Journal of Manufacturing Systems*, 9(3):233–246, 1990.

[21] D. S. Nau. Automated process planning using hierarchical abstraction. *TI Technical Journal*, pages 39–46, Winter 1987.

[22] 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.

[23] S. R. Shanker and D. G. Jansson. A generalized methodology for evaluating manufacturability. In W. G. Sullivan and H. R. Parsaei, editors, *Concurrent Engineering, Contemporary Issues and Modern Design Tools*, pages 248–263. Chapman and Hall, 1993.

[24] Jan H. Vandenbrande. *Automatic recognition of machinable features in solid models*. PhD thesis, Electrical Engineering Department, University of Rochester, 1990.

[25] H.P. Wang and R.A. Wysk. A knowledge-based approach for automated process planning. *International Journal of Production Research*, 26(6):999–1014, 1988.

[26] F.W. Wilson and P.D. Harvey. *Manufacturing Planning and Estimating Handbook*. McGraw Hill Book Company, 1963.

[27] W. Winchell. *Realistic Cost Estimating for Manufacturing*. Society of Manufacturing Engineers, 1989.

[28] M. C. Ziemke and Mary S. Spann. Concurrent engineering's roots in the world war II era. In W. G. Sullivan and H. R. Parsaei, editors, *Concurrent Engineering, Contemporary Issues and Modern Design Tools*, pages 24–41. Chapman and Hall, 1993.