

# TECHNICAL RESEARCH REPORT

## **Generating Redesign Suggestions to Reduce Setup Cost: A Step towards Automated Redesign**

*by D. Das, S.K. Gupta, D.S. Nau*

**T.R. 95-39**



*Sponsored by  
the National Science Foundation  
Engineering Research Center Program,  
the University of Maryland,  
Harvard University,  
and Industry*



# GENERATING REDESIGN SUGGESTIONS TO REDUCE SETUP COST: A STEP TOWARDS AUTOMATED REDESIGN\*

Diganta Das  
Mechanical Engr. Dept. and  
Institute for Systems Research,  
University of Maryland  
College Park, MD 20742  
diganta@cs.umd.edu

Satyandra K. Gupta  
The Robotics Institute  
Carnegie Mellon University  
Pittsburgh, PA 15213  
skgupta@isl1.ri.cmu.edu

Dana S. Nau  
Dept. of Computer Science and  
Institute for Systems Research,  
University of Maryland  
College Park, MD 20742  
nau@cs.umd.edu

## Abstract

All mechanical designs pass through a series of formal and informal redesign steps, involving the analysis of functionality, manufacturability, cost and other life-cycle factors. The speed and efficacy of these steps has a major influence on the lead time of the product from conceptualization to launching. In this paper we propose a methodology for automatically generating redesign suggestions for reducing setup costs for machined parts.

Given an interpretation of the design as a collection of machinable features, our approach is to generate alternate machining features by making geometric changes to the original features, and add them to the feature set of the original part to create an extended feature set. The designer may provide restrictions on the design indicating the type and extent of modifications allowed on certain faces and volumes, in which case all redesign suggestions generated by our approach honor those restrictions.

By taking combinations of features from the extended feature set generated above, we can generate modified versions of the original design that still satisfy the designer's intent. By considering precedence constraints and approach directions for the machining operations as well as simple fixturability constraints, we can estimate the setup time that will be required for each design. Any modified design whose setup time is less than that of the original design can be presented to the designer as a possible way to modify the original design.

---

\*This work was supported in part by NSF Grants DDM-9201779, IRI-9306580, and NSFD EEC 94-02384. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.



# 1 Introduction

Product designs typically go through a review cycle in which they are analyzed to estimate their cost-effectiveness and quality. Ideally, this design review would take into account the capabilities and costs of the production processes to be used, to allow the possibility of modifying the design to improve its manufacturability. However, it can be difficult to consider all facets of the production process in that review, particularly for complicated methods such as machining. Thus, it is sometimes not until the product enters the production cycle that process planners and machinists discover that changes to the design would improve its manufacturability—and at this point the cost of making changes can be prohibitively high.

If tools were available at the design stage to suggest design revisions to improve the manufacturability of a design, this would help to reduce both the product’s cost and lead time. This paper describes a first step toward the development of such a tool, for the domain of machined parts.

The manufacturability of a machined part depends on many factors—but one of the biggest factors is the setup time. In general, reducing the number of setups will not only reduce the time needed for manufacturing, but will also result in better machining tolerances. In this paper we describe a structured methodology for automatically generating redesign suggestions for reducing setup costs for machined parts.<sup>1</sup> The basic steps of our approach are shown in Figure 1.

In most previous work on automated redesign [13, 16, 17], the approach has been for the system to propose one or more modifications, such that each modification produces a local improvement in the manufacturability of the design. Such approaches can sometimes fail to recognize synergistic effects—situations in which a combination of modifications can improve the manufacturability of a design even though each individual modification would not improve the manufacturability if it were made by itself. Our approach is intended to overcome this drawback, by explicitly generating and considering a number of alternative designs that contain various combinations of design modifications as shown in Figure 1.

The paper is organized as follows. Following this introduction Section 2 reviews related work. Section 3 contains definitions. Sections 4 through 6 describe the details of our approach, with an example to explain how the procedure works. Finally, Section 7 includes concluding remarks and ideas for future work.

## 2 Related Work

We briefly describe some of the relevant research work in the area of automated redesign in this section. Since representing functionality of a component is also important for automated redesign, we also review some relevant work in that area.

### 2.1 Redesign of components

#### 2.1.1 Machining

Mäntylä [21] *et al.* developed a way for designers to specify alternative form features as part of the product design, through the use of what they call *feature relaxation* groups. These groups are pairs of similar geometric features which can the systems chooses the alternatives which minimize the total number of approach directions necessary for machining the part. These feature relaxation groups are similar to our feature modification operators for generating alternate features for local modification (as described in Step 2 of Figure 1). Their approach also suggests change in tolerance and technical attributes to ensure the

---

<sup>1</sup>An earlier version of our approach has been presented in [5]. The current paper goes into additional detail, and also describes several extensions and enhancements to the earlier work. In particular, the current approach uses work-holding analysis to make more realistic calculation of the number of setups, and it now calculates setup time rather than just counting the number of setups.

**Initial Step: Preprocessing.**

**Step 0a.** Get the design of the part  $P$  and the stock  $S$  from the designer. The designer may also provide restrictions indicating geometric constraints on certain portions of the design as described in Section 3.3, in which case all redesign suggestions generated by our approach will honor those restrictions.

**Step 0b.** Find all of the “primary” machining features for the original part  $P$  and the stock  $S$ . Put all these features into the set  $\mathcal{F}$ . (As discussed in Section 3.2, the primary features are sufficient for deriving all of the operation plans that will interest us.)

**Step 1: Analyze the current design.**

**Step 1a.** Find constraints on the order in which the features in  $\mathcal{F}$  can be machined, as described in Section 4.1. This set of constraints is denoted by  $C$ .

**Step 1b.** As described in Section 4.2, use  $\mathcal{F}$  to find the lowest possible setup time to machine  $P$ . Let BestTime be the time required by that operation plan.

**Step 2: Generate possible feature modifications.** For each feature  $f \in \mathcal{F}$ , try to construct new features that are similar to  $f$  but can be made using different machining operations, as described in Section 5. Let  $\mathcal{F}'$  be  $\mathcal{F}$  plus all of the new features (thus  $\mathcal{F}'$  corresponds a collection of designs similar but not necessarily identical to  $P$ ).

**Step 3. Generate and present design alternatives.**

**Step 3a.** If the set  $\mathcal{F}'$  is different from the set  $\mathcal{F}$ , then find constraints on the order in which the features in  $\mathcal{F}'$  can be machined, as described in Section 4.1. This set of precedence constraints is denoted by the set  $C'$ .

**Step 3b.** Use  $\mathcal{F}'$  to look for operation plans for designs similar to  $P$  that satisfy the geometric constraints specified in Step 0a. If plans can be found that take less setup time than BestTime (see Step 1b), then present the corresponding designs to the designer as possible redesign suggestions (see Section 6).

Figure 1: The basic steps of our approach for generating redesign suggestions.

existence of a feasible plan. However, since their approach does not consider the functional requirements of the design, there is no guarantee that the part geometry produced by the feature relaxation is compatible with the part’s intended functionality. Moreover, the objective of their work was only to minimize the number of approach directions for machining the part—and as will become clear in Section 4.2.1, this will not always reduce the setup time.

Hayes *et al.* [13] reported some advances in the direction of making redesign suggestions based on process planning knowledge. They did a protocol study to find the basic rules process planners use to reduce machining difficulty. Their approach attempts to reduce the number of setups by eliminating the setups with relatively fewer operations. As opposed to this approach we attempt to reduce the number of setups by looking at the part as a whole and at the various alternative ways of manufacturing the part.

Hayes *et al.* [14] describe a methodology for relaxing or modifying tolerances to reduce machining cost. This methodology takes into consideration the initial conditions of the stock like the surface conditions. This methodology attempts to remove setups by relaxing positional tolerances between surfaces. It does

not make changes to nominal dimensions or change shape of the part. The fixturability considerations used are elementary and may not count the number of setups required to machine the part accurately.

### 2.1.2 Other Manufacturing Processes

One of the first attempts at automated generation of redesign suggestions was made by Jakeila *et al.* [17]. Their work concentrated on automating the Boothroyd and Dewhurst [1] methods of design for assembly (DFA). The redesign suggestions are made at the design stage as and when new features are added to the design. Their system uses production rules to evaluate the design and offers suggestions for improvement as per DFA guideline. The system is limited in two major ways. First, the designer needs to create the design in terms of pre-defined feature library, which limits the freedom of the designer. Second, as the modification suggestions are made as the new features are added to the design, the order in which the design is created strongly influence the suggestions.

Hsu, Lee and Su [16] reported redesign of components for assembly from three major criteria: parallelism, assemblability and redundancy. The approach is plan-based, first the possible assembly plans are generated and then the plans are analyzed according to the criteria. They also defined some functions to modify the parts for improving on the assembly cost. They consider only a limited number of DFA guidelines and the modifications that can be suggested are minor. Moreover, in the absence of any model of the functionality requirements of the product, the modified part may not satisfy the intent of the designer.

For net shape manufacturing operations such as stamping, injection molding, and sheet metalworking, several works on manufacturability evaluation and modification have been reported in the literature. In all these cases the correspondence between design and manufacturing features are well established, and so rules could be formulated that suggest changes to individual design features in order to improve the manufacturability of those features. For example, Lazaro *et al.* have developed a methodology for finding violations of design-for-manufacturing rules for sheet-metal parts [6]. From a library of suggestions, it displays hints for modifying the design. Similar methods are also used by others [20, 28]. Complete redesigned parts are not suggested in any of these cases, but suggestions are provided for avoiding manufacturability problems detected by the domain specific manufacturability evaluator.

## 2.2 Representing Functionality

This section briefly describes recent researches on how to represent functionality of a part in its CAD model (we do not review other work in design history representation because those areas are not of direct interest to our work). In most cases the goal of the research was to find general ways to represent the functionality of designs, and so the scope of the work was very broad. In others, the research focused on a specific class of products where the features and functionality are directly coupled.

Welch and Dixon [28] developed a system for sheet metal bracket design. The only functionality they wanted to represent was load path and in the context of the particular product it was successfully accomplished. Schiebeler *et al.* [25] developed a knowledge based design assistant. This system represent functionality as a graph where the features are the nodes. The types of edges between the features depend on functional relation between the features.

ElMaraghy *et al.* [7] proposed and implemented a design scheme based on function oriented features. The functions are pre-defined into the features in the library. Functional features are also used in Schulte *et al.* [26]. Gui and Mäntylä proposed [8] a bond graph based system of assembly modelling from functional perspective. Other authors have reported product level design rationale representation systems [3, 19, 24].

Henderson [15, 27] recently reported development of a system for conceptual modeling and representation of functionality, features, dimensions and tolerances in a solid modeling system. The functionality representation scheme they used is descriptive in nature. That type of representation cannot directly be

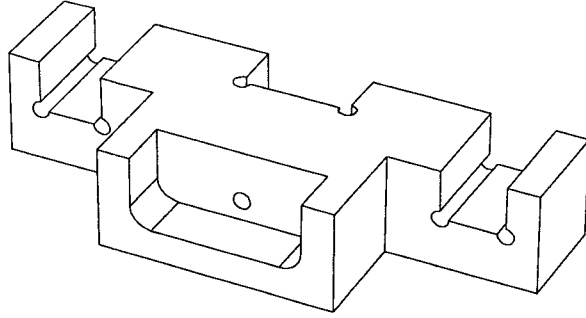


Figure 2: An example part  $P_0$ .

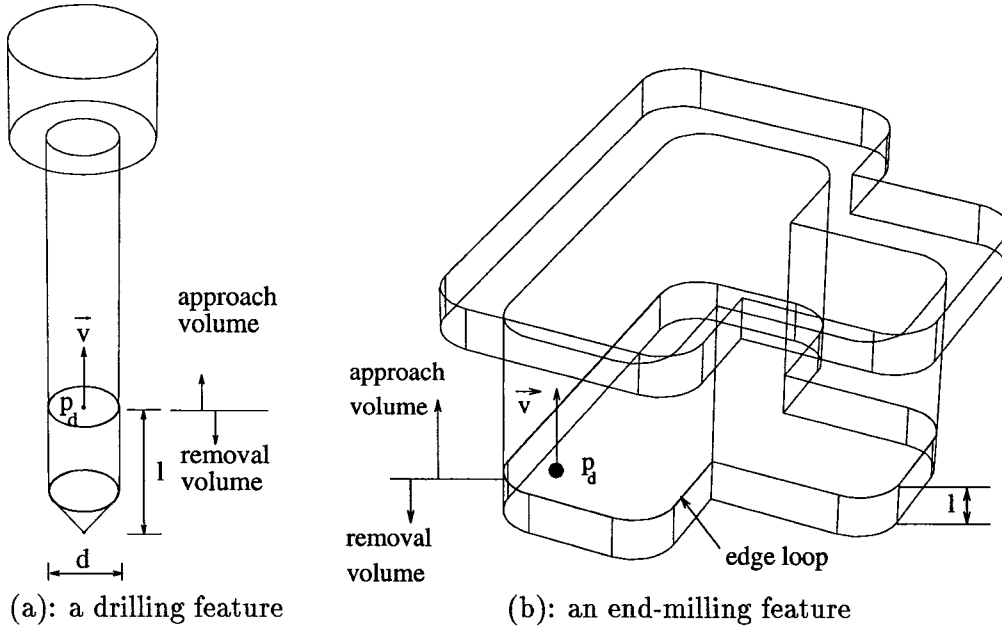


Figure 3: Examples of machining features.

used for redesign purpose with direct geometric queries. The model described by them is comprehensive and can be used as guide for future development in functional modeling of product.

### 3 Preliminaries

#### 3.1 Machining Features

A *part*,<sup>2</sup>  $P$ , is the final component created by executing a set of machining operations on a piece of *stock*,  $S$ . We assume that  $P$  and  $S$  are available as solid models. For example, Figure 2 shows an example part which we will call  $P_0$ ; this part would typically be machined from a rectangular piece of stock. For a part  $P$  and a stock  $S$ , the *delta volume* ( $S -^* P$ ), is the volume to be machined.

A *workpiece* is the intermediate object produced by performing zero or more of the operations needed to create  $P$ . The only types of operations we will consider are end milling, side milling, face milling and

<sup>2</sup>Since the research presented in this paper is a follow-up to previous work by Gupta and Nau [11], many of the preliminary definitions used here are the same as that paper.



drilling performed on a vertical machining center. For work-holding purposes, a flat jaw vise is assumed to be the only available fixturing device.

For our purposes, a *machining feature* is the portion of the workpiece affected by a machining operation. More specifically, a machining feature  $f$  will be created by some machining operation  $\text{op}(f)$ , using a cutting tool  $\text{tool}(f)$ . To perform the machining operation, one sweeps the tool along some trajectory that is characterized by some set of parameters  $\text{param}(f)$ . The *removal volume* is the portion of this swept volume in which the cutting tool is actually capable of removing material. The *accessibility volume* is the remaining portion of the tool swept volume. The *approach face* separates the removal volume from the accessibility volume. The *effective removal volume* is the intersection of the removal volume with the stock. Below are two examples:

- Suppose we want to drill the hole  $h$  shown in Figure 3(a). Then  $\text{op}(h)$  will be drilling. To create  $h$ , we will sweep a drilling tool  $\text{tool}(h)$  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,  $\text{param}(h)$  is the set  $\{p_d, \vec{v}, d, l\}$ . The accessibility volume and the removal volume are as shown in the figure.
- Suppose we want to mill the pocket  $p$  shown in Figure 3(b). Then  $\text{op}(p)$  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,  $\text{param}(p)$  is the set  $\{p_d, \vec{v}, e, l\}$ . The accessibility volume and the removal volume are as shown in the figure.

### 3.2 Primary Features, and Feature-Based Models

A *primary feature* for a given part  $P$  and stock  $S$  is a machining feature that is minimal with respect to  $S$  and maximal with respect to  $P$ . Figure 5 shows an example; for a detailed definition the reader is referred to [11, 12]. As described in [9, 23], the reason why we are interested in primary features is that they are sufficient for deriving all of the machining operations we might wish to perform to create the part  $P$ . In particular, for every machining operation we might want to use in creating  $P$ , the operation will create either a primary feature or a truncation of a primary feature.

Given a part  $P$  and stock  $S$ , we will let  $\mathcal{F}$  denote the set of *all* primary features for  $P$  and  $S$ . In [23, 10], we describe an algorithm that, given  $P$  and  $S$ , will automatically find  $\mathcal{F}$ . For example, in the case of the part  $P_0$  shown in Figure 2,  $\mathcal{F}$  contains 30 features, some of which are shown in Figure 4. In particular,  $h1, h51, h52$  are drilling features;  $s3, s4, s6$  are end-milling features; and  $s11, s12, s14$  are side milling features. Figure 4 also shows that  $\mathcal{F}$  may contain several different primary features corresponding to the same portions of  $P$ .

A *Feature Based Model* (FBM) for  $P$  is any irredundant subset  $F \subseteq \mathcal{F}$  such that  $P$  can be produced from  $S$  by removing the features in  $F$ . For example, the following sets of features from Figure 4 are two feature based models for  $P_0$ :

$$\begin{aligned} \text{FBM1} &= \{s1, s2, s3, s4, s5, s6, h1, h41, h42, h51, h52, h61, h62\}; \\ \text{FBM2} &= \{s1, s2, s3, s5, s7, s8, h1, h81, h82, h51, h52, h71, h72\}. \end{aligned}$$

Each operation plan for creating  $P$  from  $S$  is a sequence of machining operations. In general, there may be several different operation plans capable of creating  $P$  from  $S$ . We will not want to consider all of these plans (for example, we will not be interested in plans for which some of the operations are redundant). As described in [9, 12], each operation plan of interest to us corresponds to an FBM for  $P$ , in the sense that each machining operation in the plan will create either some feature in the FBM or a truncation of some feature in the FBM. In order to compute setup times, we will never actually compute truncations of the features in  $\mathcal{F}$ , because we can determine how many setups are needed directly from the FBMs.

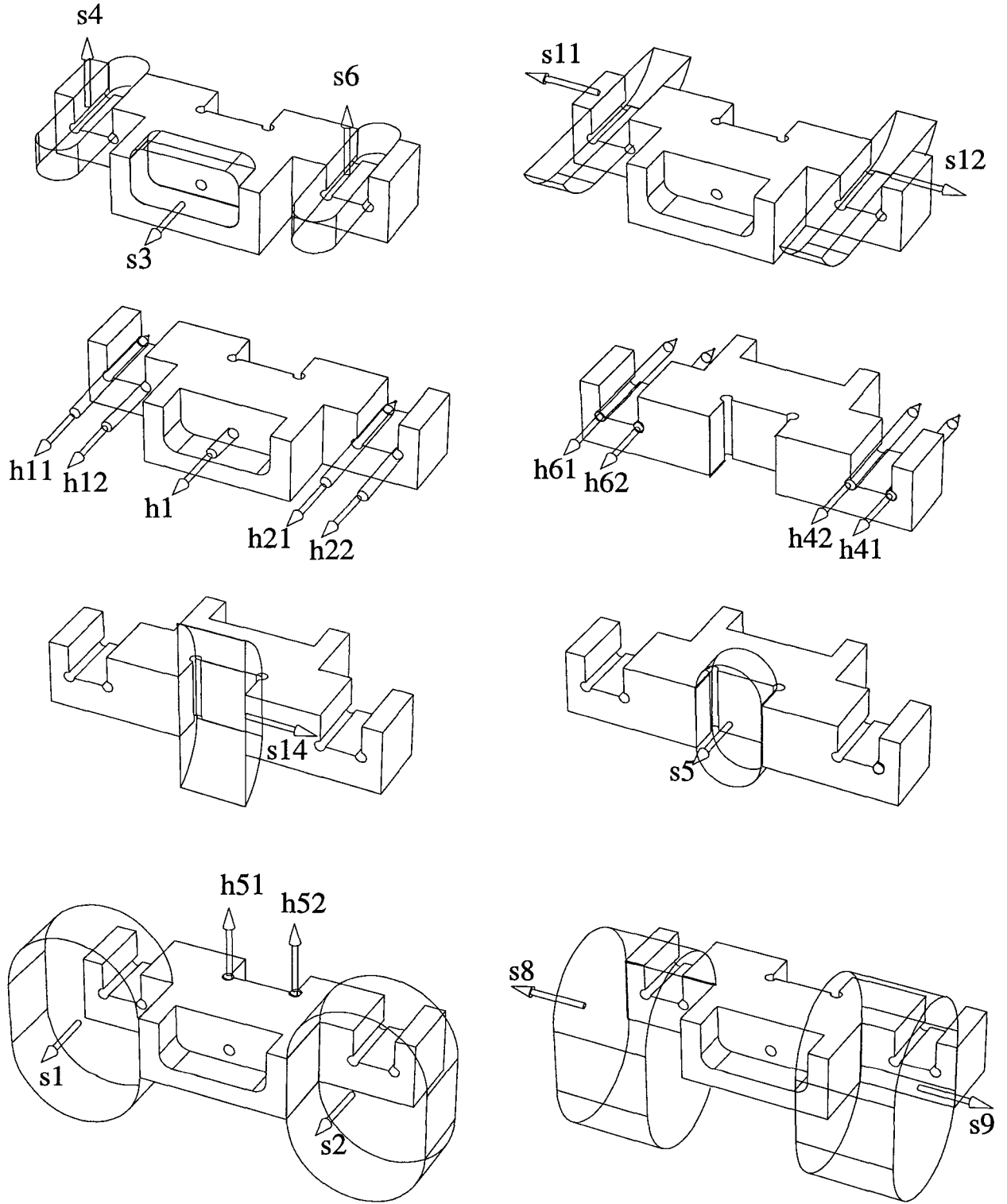


Figure 4: Examples of some of the primary machining features for the part  $P_0$ .

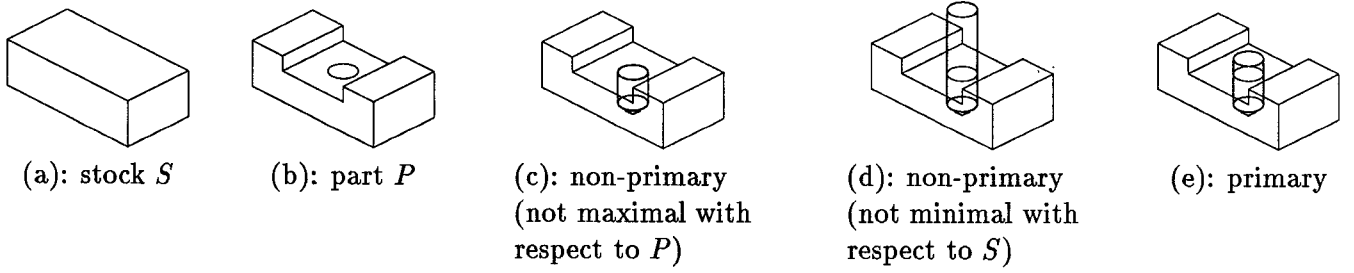


Figure 5: Example of primary and non-primary features (reproduced from [11]).

### 3.3 Geometric Constraints on the Design

In proposing redesign suggestions, we will want to try to ensure that the modified design still achieves the functionality intended by the designer. However, the functional requirements for a design can be quite complex and disparate in nature, and we are not interested in developing a detailed scheme for representing them. Instead, our approach is based on the idea of representing various geometric constraints arising from the intended functionality, without trying to represent the functionality itself.

For example, machined parts typically are components of larger assemblies, and many of the design constraints will involve how the part interacts with other portions of the assembly [22]. Thus, many of the constraints associated with a machined part will correspond to regions of space in which it mates with other portions of the assembly.

**Constraint volumes.** To represent such constraints, we will ask the designer to specify *constraint volumes* and various geometric constraints associated with those volumes, to provide limits on what kinds of geometric variations in the part are permissible. The intent is that both the original design and all possible modified versions of the design should satisfy these constraints.

Each constraint volume  $l$  is a specific volume of space such that (with the possible exception of edge blends), neither the design nor any modified version of the design should intersect with  $l$ . Currently, the constraint volumes are restricted to be linear sweeps of non-self-intersecting planar edge-loops made up of linear and circular edges.

The geometric constraints associated with  $l$  specify how various faces and edges of the design should or should not touch  $l$ . The details of this scheme are presented below.

**Constraints on faces.** If  $f$  is a planar face on the constraint volume which is co-incident with some face on the boundary of the part, then the designer can associate with  $f$  either of the following types of constraints (examples are given later in Figure 7 and Table 1):

1. For any portion  $p$  of  $f$ , the designer may specify that the design (and any modified version of it) must include  $p$  as part of its boundary.
2. The designer may specify minimum and maximum areas for the surface of intersection  $u = f \cap^* b$ , where  $b$  is the boundary of the part  $P$  or any modified version of  $P$ . The designer may also specify a circular region of space within which  $u$ 's centroid must be located.

**Constraints on edges.** Edges on the constraint volume control the type and degree of contact of the constraint volume with the part under consideration. As we will see in Section 5, frequently the modifications done to the machining features involve machining a feature from a different direction. In these cases, some of the edges present in a machining feature will become blended to have a corner radius.

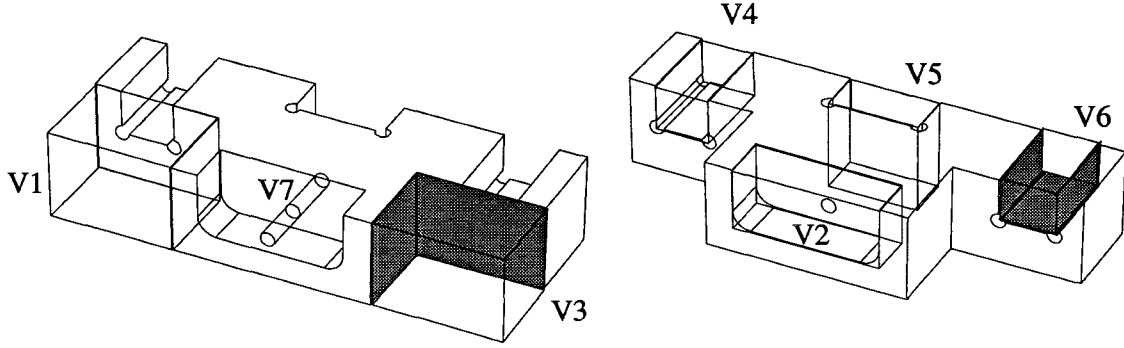


Figure 6: The volumes for the example part on which geometric constraints are put

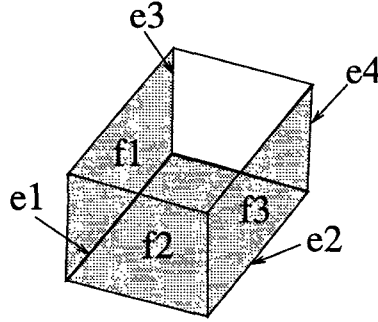


Figure 7: Details of the geometric constraints put on volume V6 shown in Figure 6

For controlling the types of modifications we will allow the designer to specify the following constraints on any linear convex edge  $e$  of the constraint volume that is not completely exterior to the stock:

1. *No-contact constraint*:  $e$  is not allowed to touch the boundary of  $P$  (where  $P$  is the design or any modified version of it).
2. *Edge-clearance constraint*:  $e$  can touch a face of  $P$ , but is not allowed to intersect any edge of  $P$  except possibly at a single point.
3. *Edge-contact constraint*:  $e$  must coincide at least partially with some edge of  $P$ , i.e., there must be an edge of  $P$  whose intersection with  $e$  is a line segment.

In addition to the above conditions, if a linear convex edge  $e$  of the constraint volume is at least partially coincident with an edge of  $P$ , then the designer can specify  $e$  to be blendable and assign a maximum blending radius.

Throughout the rest of this paper, we will assume that the designer has associated constraint volumes with the part  $P_0$  as shown in Figure 6. For brevity, we will not describe the geometric constraints associated with most of these constraint volumes. However, Table 1 shows the geometric constraints for the constraint volume V6 shown in Figure 7.

Table 1: Geometric constraints on volume V6.  
*Geometric Constraints on Faces*

face	Center and radius (mm) of the region in which the centroid must lie	Min area (mm <sup>2</sup> )	Max area (mm <sup>2</sup> )
f1	(40,6.25,6.25) r=7.5	200	300
f2	(47.5,6.25,0) r=2.5	200	400
f3	(55,6.25,6.25) r=7.5	200	300

*Geometric Constraints on Edges*

edge	Type of constraint
e1	none or edge clearance
e2	none or edge clearance
e3	edge contact
e4	edge contact

### 3.4 Admissible Feature Sets

Later in this paper we present a method for automatically generating suggestions for how to change the design of a part. This method works by proposing changes to some of the features that appear in FBM's for the part. These changed collections of features will no longer be FBMs, because they will no longer describe the exact geometry of the original design. However, we will still require them to satisfy the geometric constraints described in Section 3.3. In particular, they must be *admissible feature sets*. An admissible feature set (AFS) is any subset  $\Phi \subseteq \mathcal{F}$  that satisfies the following properties:

1. The part created by subtracting the features in  $\Phi$  from the stock satisfies all the geometric constraints set by the designer;
2. No part created by subtracting the features of any proper subset  $H \subset \Phi$  from the stock will satisfy the geometric constraints.

If the geometric constraints specified by the designer are consistent with the geometry of the part design, then every FBM (as defined in Section 3.2) for  $P$  will satisfy the geometric constraints, and will thus be an AFS. For the example part  $P_0$ , the FBMs described in Section 3.2 are also AFS. Ordinarily, every AFS for  $P$  should also be an FBM for  $P$ , i.e., we should be able to create  $P$  by starting with  $S$  and removing the features in the AFS. There are two circumstances, when this may not be true.

1. The designer did not put all the geometric constraints necessary for the part. If this problem is not corrected, then the modifications suggested by the system may violate the designer's intent.
2. There is some removal volume which is not important and the designer did not put any constraints relating to that volume. In this case, the design should be modified to eliminate that removal volume.

## 4 Analyzing the design

This section describes Steps 1a and 1b of Figure 1, and illustrates how they would be carried out on the part  $P_0$ .

### 4.1 Step 1a: Finding Precedence Constraints

Due to various types of interactions among the features used to machine a part, the features cannot be machined in any arbitrary order. Instead, these interactions introduce precedence constraints requiring that some features be machined before or after other features. We are interested in finding these precedence constraints among features in  $\mathcal{F}$ . The number of setups (and hence setup time) required to machine the part will depend on them (see Section 4.2.1). The only feature pairs among which we will look for precedence constraints are those which are intersecting but do not have the same effective removal volume. If the features have the same effective removal volume then both of those will not be in the same AFS.

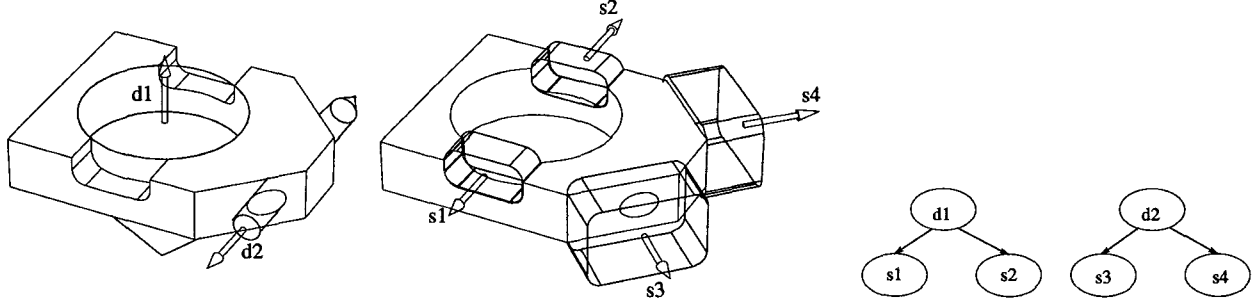


Figure 8: A part in which some of the features must be machined before others and the precedence constraints among them.

As an example, Figure 8 shows a part in which the slot-hole interactions create precedence constraints for machining of that part. The large vertical hole d1 must precede the two end-mill features s1 and s2 on its side for proper drill engagement. Also, to get flat entry and exit face while machining the drilling feature d2, the horizontal hole d2 must precede the end-mill features s3 and s4.

More generally, a pair of features  $f$  and  $f'$  will have precedence constraints under the following conditions:

1. If the accessibility volume of feature  $f'$  intersect with the removal volume of feature  $f$ , then  $f$  has to be machined before  $f'$ .
2. Precedence constraints also arise from preferred manufacturing practices. Some examples are given below; more such cases are enumerated in [2, 4].
  - (a) if  $f$  is an end-milling feature with at least one side open and  $f'$  a drilling feature which has the same approach direction and its removal volume intersect with the removal volume of  $f$ , then machine  $f$  before  $f'$ ;
  - (b) if  $f$  is a side-milling feature and  $f'$  is a drilling feature and the removal volumes of the two features intersect, then machine  $f$  before  $f'$ ;
  - (c) if two drilling features  $f$  and  $f'$  are collinear and their removal volumes intersect, then drill the smaller of the two first.

The procedure to find the precedence constraints is straightforward. We simply check, for each pair of features which has volumetric intersection and do not have the same effective removal volume, whether the above conditions hold.<sup>3</sup>

## 4.2 Step 1b: Finding the Minimum Setup Time

In the previous section, we found precedence constraints among all of the features in  $\mathcal{F}$ . However, to machine the part, one will not machine all of these features. Instead, one will machine some subset  $\Phi \subseteq \mathcal{F}$

<sup>3</sup>One limitation of this approach is that since we are looking for precedences on the primary features rather than truncating them to get the features that would actually be used in operation plans, in some cases we will find precedence constraints that are not actually needed in any operation plan. This can occasionally cause our approach to overestimate the number of setups that will be required for machining a part. To avoid this problem, Gupta [12] gives a more detailed procedure for finding and assigning precedence constraints among features, by looking for precedence constraints within each FBM. In this paper we do not use Gupta's procedure, for the following reasons. First, using Gupta's procedure within our framework would greatly increase the amount of computation required because we may potentially generate a very large number of AFSs during the redesign suggestion generation procedure. Second, our method of finding precedence constraints will only rarely overestimate the number of setups needed, and usually not by much.

that satisfies all the geometric constraints on  $P$ . As defined in Section 3.4,  $\Phi$  is called an Admissible-Feature-Set (AFS). The number of setups required to machine  $\Phi$  is determined primarily by the approach directions of the features and the precedence constraints among them. The number of setups also depends on the presence of required types of faces in the intermediate workpieces for holding the workpiece on the machine table and for probing the workpiece.

The minimum number of setups needed to create  $P$ , is the minimum, over every AFS  $\Phi$ , of the number of setups required for  $\Phi$ . Potentially there may exist a large number of AFSs for  $P$ , and the problem is how to compute the minimum number of setups required to machine  $P$  without enumerating all the AFSs. To do this, we use the procedure ANALYZE-DESIGN described below which computes the best possible setup time for the part. ANALYZE-DESIGN is a branch-and-bound procedure that finds AFSs, one by one, and computes the number of setups (and setup time) for the ones that appear promising.

In addition to the part  $P$  and stock  $S$ , ANALYZE-DESIGN takes the following arguments: the set of features  $\mathcal{F}$ , the set of precedence constraints  $C$  among the features, the set  $L$  of the functional volumes specified by the designer, the current best setup time BestTime (which is initially set to  $\infty$ ), a set of machining features  $\mathcal{G}$  from which an AFS needs to be built and a partial AFS  $G$  (which is initially empty). The values for BestTime and  $G$  are revised by ANALYZE-DESIGN during its recursive calls.

**procedure** ANALYZE-DESIGN( $L, \mathcal{G}, G, \text{BestTime}, P, S, C$ )

1. **Pruning Step:** If  $n * t_s > \text{BestTime}$ , (where  $n$  is the number of approach directions of the features in  $G$  and  $t_s$  is the setup time for each setup) then return  $\infty$ , because  $G$  will not result in an AFS which takes lower setup time than BestTime.
2. **Redundancy Test:** Let  $d = S \cap^* \mathcal{L}$ , where  $\mathcal{L}$  is the union of all the constraint volumes  $l \in L$  after blending all the blendable edges. Thus,  $d$  is the smallest possible delta volume that that could result from any possible redesign of  $P$ . If there is a  $g$  such that  $d \cap^* (\cup^*(G - \{g\})) = d \cap^* (\cup^*G)$ , then  $g$  is redundant because it doesn't remove any portion of  $d$  that is not removed by the other features. Thus return  $\infty$ .
3. **Feasibility Test:** If  $G$  will not result in an AFS which creates a valid part, then return  $\infty$  (this step is described in Appendix B).
4. **Goal Test:** If  $G$  satisfies all of the geometric constraints on the volumes in  $L$  (as is described in Appendix B), then  $G$  is an AFS, so do the following:
  - (a) If  $G$  is not an FBM for  $P$  and  $S$ , then as described in Section 3.4, there is a problem with the constraints specified by the designer—and the designer should be notified of this and given the choice to either edit the geometric constraints or to modify the design itself before proceeding further.
  - (b) Otherwise, use the following procedure (which is described in Section 4.2.1) to find the machining sequence for the features in  $G$  that requires the least setup time:  
Let  $T = \text{FIND-BEST-SETUP-TIME}(G, C, \infty, 0)$ .  
Return  $T$
5. **Recursion Step:** Pick a feature  $g$  from  $\mathcal{G}$ , and do the following:
  - (a)  $\text{BestTime} = \min(\text{BestTime}, \text{ANALYZE-DESIGN}(L, \mathcal{G} - g, G \cup g, \text{BestTime}, P, S, C))$
  - (b)  $\text{BestTime} = \min(\text{BestTime}, \text{ANALYZE-DESIGN}(L, \mathcal{G} - g, G, \text{BestTime}, P, S, C))$
  - (c) Return BestTime

### 4.2.1 Estimating Setup Time

Once we have an AFS, it is a specific set of machining features all of which needs to be machined to get the final part. Each feature  $f$  has a specific approach direction  $\vec{v}(f)$ , and some of these features may have some precedence constraints among them. The number of setups needed to machine the AFS will be the minimum, over all feature machining sequences satisfying the precedence and work-holding constraints of the number of setups required by the operation sequence.

In this paper vise clamping is assumed to be the only work-holding method available. As per Wilson's [29] handbook, for a flat-jaw vise, the fixturing time is between 0.8 to 1.2 minutes depending on the weight of the part. In a vertical machining center, additional time is needed to probe the workpiece. We add 1.0 minute to the fixturing time to account for probing time. Under these conditions:

$$\text{setup time} = n \times t_s,$$

Where  $t_s$  is the average setup time for vise clamping, and  $n$  is the minimum possible number of setups. We take  $t_s$  to be equal to 2 minutes for each setup.

The branch and bound procedure FIND-BEST-SETUP-TIME described below estimates the minimum setup time required to machine a given AFS  $\Phi$ . The procedure HOLDING-ANALYSIS called by FIND-BEST-SETUP-TIME analyses the feasibility of machining a collection of machining features in a setup.

The procedure FIND-BEST-SETUP-TIME takes as argument a set of features  $B$  which are to be put in valid setups, the set of precedence constraints among the features in  $B$  and the setup time  $t$ . Initially the set of features is the AFS  $G$  found by the procedure ANALYZE-DESIGN and the setup time is nil. The argument  $T$  (initially  $\infty$ ) is used by FIND-BEST-SETUP-TIME to hold the best setup time it has seen in any of the setup sequences it has explored so far.

**procedure** FIND-BEST-SETUP-TIME( $B, C, T, t$ )

1. If  $t \geq T$ , then return  $T$ , because the setup time is not better than the best solution so far. This condition prevents unpromising setup sequences from being investigated further.
2. If  $B = \emptyset$ , then there are no remaining features, so return  $t$ . Otherwise, do the steps below.
3. Let READY be the set of all features in  $B$  that have no predecessors.
4. Let  $V$  be the set of all approach directions of features in READY (i.e.,  $V = \{\vec{v}(f) : f \in \text{READY}\}$ ).  $V$  contains the approach directions from which the next setup can be machined.
5. For every approach direction  $\vec{v} \in V$ , do the following:<sup>4</sup>
  - (a) Let  $H$  be the set of all features  $f \in B$  such that
    - i.  $f$  has  $\vec{v}$  as its approach direction;
    - ii. either  $f$  has no predecessor in  $B$ , or all predecessors in  $B$  have  $\vec{v}$  as their approach direction.

Note that all of these features can be machined in the same setup if the fixturability conditions permit.
  - (b) Let  $W = S - * ((\cup^* F) - * (\cup^* B))$ .  
 $W$  represents the current workpiece, i.e., the workpiece after machining the features already removed from  $B$  during its recursive calling sequence (see Step 5(c)ii below).

---

<sup>4</sup>For efficiency considerations, we pick  $v$  in decreasing order of the number of features in READY whose approach direction is  $v$ .



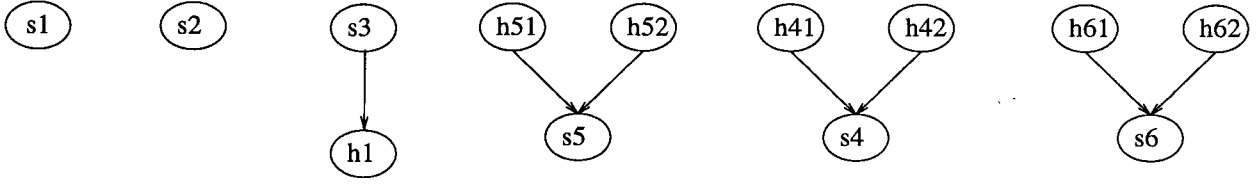


Figure 9: Precedence constraints among the features of the AFS 1 (described in Section 3.4)

(c) If  $\text{WORKPIECE-PROBE}(W, \vec{v})$ , then

i.  $\mathcal{K} = \text{HOLDING-ANALYSIS}(W, \vec{v}, H, C)$

(Each element of the set  $\mathcal{K}$  returned by  $\text{HOLDING-ANALYSIS}$  is a set of features  $K \subseteq H$  that can be machined in one setup.)

ii. Remove from  $\mathcal{K}$  any set  $K'$  that is a proper subset of some other set  $K \in \mathcal{K}$

For each  $K \in \mathcal{K}$ ,<sup>5</sup>

$T = \min(T, \text{FIND-BEST-SETUP-TIME}(B - K, C - C', T, t + t_s))$ , where  $C'$  is the set of all precedence constraints in  $C$  that involve at least one feature in  $K$ .

In Step 5c of the procedure  $\text{FIND-BEST-SETUP-TIME}$  we check the workpiece geometry to find if it has faces and features which allow locating the workpiece on a CNC machining center (described in Appendix C.1). We proceed with the workholding analysis only if the workpiece has that property, otherwise a different setup is chosen.

In Step 5(c)i,  $\text{FIND-BEST-SETUP-TIME}$  uses a procedure called  $\text{HOLDING-ANALYSIS}$  to find alternative sets of features  $\mathcal{K}$  that can be machined in one setup. Procedure  $\text{HOLDING-ANALYSIS}$  is described in Appendix C.  $\text{HOLDING-ANALYSIS}$  assumes that a vise is the only available fixturing device—but we are developing procedures for use with other types of work-holding devices (such as clamping), and we intend to use these procedures to augment the set  $\mathcal{K}$ . Since  $\text{HOLDING-ANALYSIS}$  assesses fixturability independent of the rest of the analysis, it will be straightforward to incorporate these procedures into our approach.

#### 4.2.2 Result of the algorithm on an example part

For the part  $P_0$ , there are several AFSs. Two such AFSs are AFSs 1 and 2, mentioned in Section 3.1.

Among all AFSs created by the algorithm, AFS 1 can be machined in the lowest number of setups. AFS 1 has only three approach directions—but due to the precedence constraints among these features (see Figure 9), the minimum number of setups required to machine AFS 1 is four. So it will need a minimum of 8 minutes as setup time, as the time for each setup with vise jaw is taken to be 2 minutes. In this case there are no problems arising from the fixturability point of view.

Which AFSs are found by the procedure  $\text{ANALYZE-DESIGN}$  will depend on the order in which features are placed in the AFS being generated. Any time a partial AFS  $G$  is generated whose setup time exceeds best setup of the AFSs found so far, the algorithm will discard  $G$ . Since AFS 1 has the lowest setup time of any possible AFS for the part, it will always be generated—but since AFS 2 needs a longer setup time, it may or may not be fully generated, depending on whether algorithm starts generating it before or after generating AFS 1.

<sup>5</sup>For efficiency considerations, we pick  $K$  in decreasing order of cardinality among all the  $K \in \mathcal{K}$

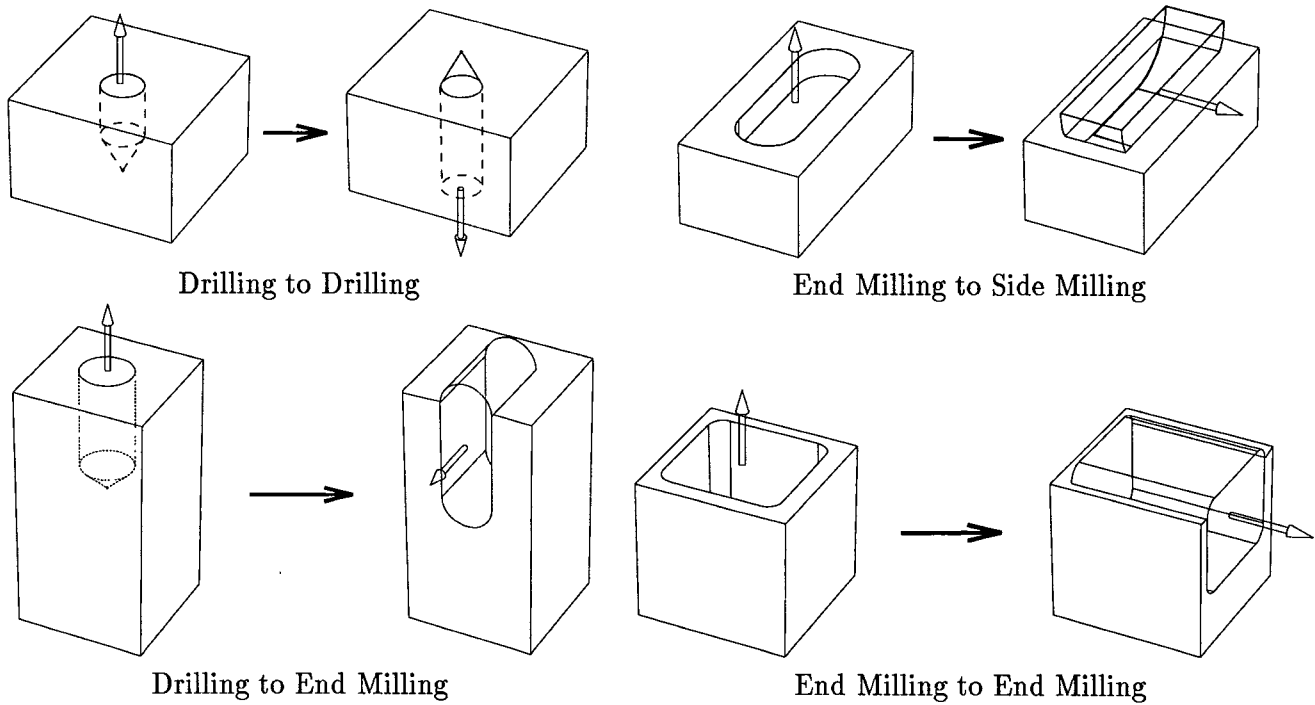


Figure 10: Examples of feature modification.

## 5 Step 2: Machining Feature Modifications

### 5.1 General Procedure

In Section 4 we presented a method for finding the minimum setup cost for machining a part. We are interested in improving on this setup cost, by considering modifications to the geometry of the existing design. This section describes in detail the Step 2 of Figure 1.

To avoid having to go back to the conceptual design stage, we will only consider local modifications on the machining features already found in the original design. Since the number of setups will depend on the approach directions for the features and the precedence constraints among them, our objective is to modify some of the features in such a way as to allow them to be machined from different approach directions. We want to ensure that if the old feature is replaced by the new feature in the part design, that will not adversely affect the functionality of the rest of the part.<sup>6</sup>

Our basic approach involves the use of feature modification operators that, given a machining feature as input, produce alternative features that are similar but not identical to the input feature. As illustrated in Figure 10, these feature modification operators are of two different types:

1. perform the same machining operation from a different direction;
2. use a different machining operation.

We use these feature modification operators in the manner described below:

#### **Procedure GENERATE-MODIFIED-FEATURE-SET**

<sup>6</sup>As described in the Section 4.2.1, there could be requirement of more setups due to work-holding constraints, but our modifications do not address that.

1. Initially set  $\mathcal{U} = \emptyset$  :  $\mathcal{U}$  is the set of features generated by the feature modification operators that are not acceptable to the designer as possible alternatives (see Step 3(a)i).
2. Initially set  $\mathcal{F}' = \mathcal{F}$
3. For every feature  $f \in \mathcal{F}$  do:
  - (a) Apply all feature modification operators that are applicable to  $f$ , to get new features (there can be more than one feature created by one operator). For each feature  $f'$  that is created, do:
    - i. If replacing  $f$  by  $f'$  in the original part will result in some violation of the geometric constraints which cannot be remedied by any means, then discard  $f'$ .
    - ii. Otherwise, if  $f'$  is not already in the set  $\mathcal{F}'$  or in the set  $\mathcal{U}$  (which would happen if the new feature were previously generated by modifying some other feature, then either accepted or rejected by the designer), then display that feature to the designer as a possible alternative to  $f$ .<sup>7</sup> If the designer accepts  $f'$  as a possible alternative, then  $\mathcal{F}' = \mathcal{F}' \cup f'$ . If the designer does not accept the feature  $f'$  as a possible modification then put  $f'$  in the set  $\mathcal{U}$ .

If the designer does not find the new feature  $f'$  to be acceptable in Step 3(a)ii, this means that in some way or another, the new feature violates the designer's intent. One way that this can happen is if the geometric constraints specified by the designer were not sufficient to represent all of the functional requirements that the designer had in mind. In that case, the designer can go back and modify the geometric constraints, and restart the analysis procedure.<sup>8</sup>

These steps extend the feature set  $\mathcal{F}$ , to create a new set  $\mathcal{F}'$ . In addition to the features of the original part,  $\mathcal{F}'$  contains those created by the feature modification operators that do not violate the geometric constraints<sup>9</sup> and are acceptable to the designer as possible local modifications.

## 5.2 The feature modification operators

We currently have defined seven different feature modification operators. Each operator takes as inputs the part  $P$ , the stock  $S$  and the feature  $f$  to be modified. If  $f$  does not satisfy the operator's applicability conditions, then the operator produces no output. If  $f$  does satisfy the applicability condition, then the operator will examine the parameter set used to describe  $f$ , and use this information to generate one or more alternatives for  $f$ .

As discussed in Section 5.1, we will often want to generate features having different approach directions from  $f$ , but we need to have some way of limiting the number of approach directions from which modified features are to be machined. We define the set  $D = D_1 \cup D_2$ , where

$$\begin{aligned} D_1 &= \{\text{the approach directions of all features in } \mathcal{F} \text{ and the directions opposite to those}\}; \\ D_2 &= \{v : v \text{ is a vector perpendicular to a planar face of } P \text{ or } S \text{ and pointing outward}\}. \end{aligned}$$

Each operator will only generate features which can be machined from approach directions in  $D$ .

---

<sup>7</sup>The purpose of this step is not to suggest immediately to the designer that the designer should replace  $f$  with the new feature, but simply to find out whether the new feature might possibly be acceptable to the designer. If it is, then putting it into  $\mathcal{F}'$  makes it a possible candidate for generating redesign suggestions later on, as described in Section 6.

<sup>8</sup>Ultimately, we would like to provide ways whereby the analysis procedure can take up where it left off, incorporating the modified geometric constraints into its analysis. This is a topic for future work.

<sup>9</sup>This does not mean that a redesign suggestion generated by using this feature will not violate any geometric constraint. We cannot check for all possible violations of geometric constraints at this level. We can check only some types of violation, which are detailed in Section 4.2

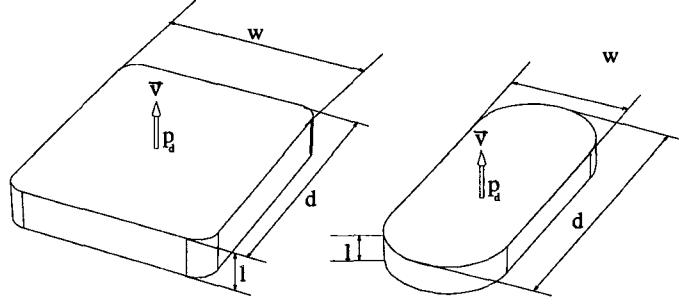


Figure 11: Examples of end milling features which can be modified or generated.

The following paragraphs describe the alternatives generated by each operator. Of these alternatives, the operator will discard every alternative  $f'$  that splits  $P$  into more than one piece; i.e. it will discard  $f'$  if the solid  $((P \cup^* f) -^* f')$  is not a single manifold solid.

Now we give brief descriptions of each operator we defined. These are named in the format  $\mathcal{O}_x^y$ . Each operator  $\mathcal{O}_x^y$  takes as input a feature of type  $x$ , and produces features of type  $y$ . The details of some of these operators are presented in the Appendix A.

#### Operators for modifying drilling features:

$\mathcal{O}_d^d(P, S, f)$ : This operator is only applicable if  $f$  is a drilling feature that corresponds a blind hole; i.e.,  $f$ 's conical bottom must have a non-empty intersection with  $P$ . If applicable, this operator produces a through hole, which can be machined from the opposite direction.

$\mathcal{O}_d^e(P, S, f)$ : This operator is applicable to all drilling features. It generates an end milling feature  $f'$  for each approach direction in  $D$  which is perpendicular to the axis of  $f$ .

$(\mathcal{O}_d^e)(P, S, f)$ : This operator is applicable to all drilling features. It generates one milling feature  $f'$  for each approach direction in  $D$  which is perpendicular to the axis of  $f$ .

**Operators for modifying end milling features:** The end milling features on which these operators are applicable (and the end milling features which are created by these operators) limited in the scope of their geometry.<sup>10</sup> Only those end milling features for which the edge loop can be defined completely by the following set of parameters are possible to modify or generate.

$$\{p_d(f), \vec{v}(f), d(f), l(f), w(f)\}$$

The only types of end milling features which can be modified or generated are ones with rectangular edge-loop with the corners blended with circular edges of same radius. If two blended radii interconnect it has to create a complete semi-circle. Figure 11 shows the end milling features on which these operators are applicable.

$(\mathcal{O}_e^e)(P, S, f)$ : This operator creates end milling features from different direction from an end milling feature. The approach directions from which the new end milling features are created are limited to the ones perpendicular to the planar faces of the end milling feature. This is followed so that the geometric constraints on those faces can be maintained.

<sup>10</sup>The part design is not limited to these types of features, that may contain generalized milling features. Our methodology will not be able to modify those features or generate generalized milling features from other features.

$(O_e^s)(P, S, f)$ : This operator is similar to the previous operator, only difference being that the new feature(s) are side milling features. The approach directions from which the side milling features are created are parallel to the planar faces in the original feature.

### Operators for modifying side milling features:

$(O_e^e)(P, S, f)$ : This operator creates one or more end milling features from a side milling feature. The approach directions from which the new end milling features are created are limited to the ones perpendicular to the planar faces of the side milling feature. This is followed so that the geometric constraints on those faces can be maintained.

$(O_s^s)(P, S, f)$ : This operator is similar to the previous operator, only difference being that the new feature(s) are side milling features. The approach directions from which the side milling features are created are parallel to the planar faces in the original feature.

## 6 Step 3: Generating design alternatives

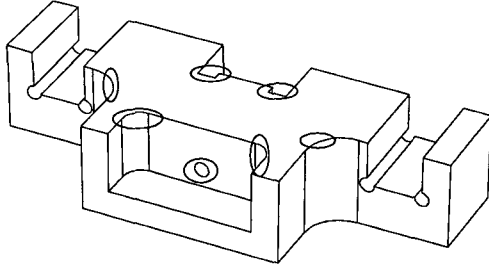
In order to find design alternatives in Step 3 of Figure 1, we will be interested in finding each AFS  $\Phi$  such that the part created by subtracting  $\Phi$  from the stock  $S$  is a single manifold solid that will require less setup time to machine than  $P$ . Each such  $\Phi$  we will call an Redesigned Admissible Feature Set RAFS. Each RAFS will correspond to a potential modified version of  $P$ . If we cannot find an RAFS that means that our approach cannot find a redesign suggestion for the original part.

After the precedence constraints are found (see Section 4.1) among the features in the extended feature set  $\mathcal{F}'$  we will attempt to extract possible alternative parts. The procedure GENERATE-REDESIGN-SUGGESTIONS generates a set  $\mathcal{R}$  of possible RAFSs which is similar (but not identical) to the procedure ANALYZE-DESIGN of Section 4.2. BestTime is previously calculated by the procedure ANALYZE-DESIGN.

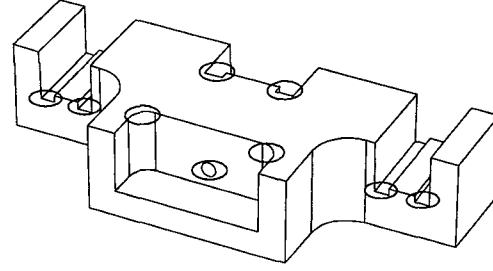
In addition to the part  $P$  and stock  $S$  GENERATE-REDESIGN-SUGGESTIONS takes the following arguments: the extended feature set  $\mathcal{F}'$  and the precedence constraint among the features  $C'$ , the set  $L$  of the functional volumes specified by the designer, a constant BestTime calculated previously by the procedure ANALYZE-DESIGN in Section 4.2, a set of machining features  $\mathcal{G}$  from which an AFS needs to be built and a partial AFS  $G$  (which is initially empty).

**procedure** GENERATE-REDESIGN-SUGGESTIONS( $L, P, S, \mathcal{G}, G, \text{BestTime}, C'$ )

1. **Pruning Step:** If  $n * t_s \geq \text{BestTime}$  then return  $\emptyset$ , (where  $n$  is the number of approach directions of the features in  $G$  and  $t_s$  is the setup time for each setup) because  $G$  will not result in an AFS which takes lower setup time than BestTime.
2. **Redundancy Test:** Let  $d = S \cap^* \mathcal{L}$ , where  $\mathcal{L}$  is the union of all the constraint volumes  $l \in L$  after blending all the blendable edges. Thus,  $d$  is the smallest possible delta volume that that could result from any possible redesign of  $P$ . If there is a  $g$  such that  $d \cap^* (\cup^*(G - \{g\})) = d \cap^* (\cup^* G)$ , then  $g$  is redundant because it doesn't remove any portion of  $d$  that is not removed by the other features. Thus return  $\emptyset$ .
3. **Feasibility Test:** If  $G$  will not result in an AFS which creates a valid part, then return  $\emptyset$  (this step is described in Appendix B).
4. **Goal Test:** Otherwise, if  $G$  satisfies all of the geometric constraints on the volumes in  $L$  (as it is described in Appendix B), then  $G$  is an AFS, so do the following:



(a) :  $P_0^1$ , a modified version of the part  $P_0$  with the modifications highlighted.



(b) :  $P_0^2$ , another modified version of the part  $P_0$  with the modifications highlighted.

Figure 12: Two modified versions of the part  $P_0$

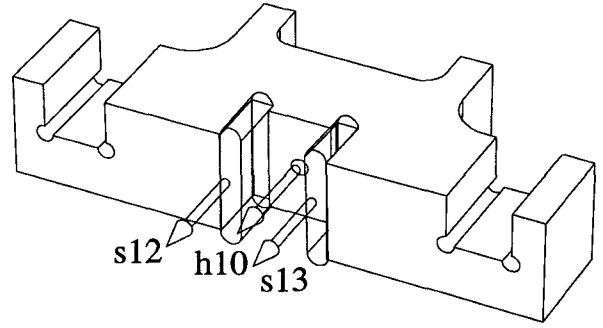
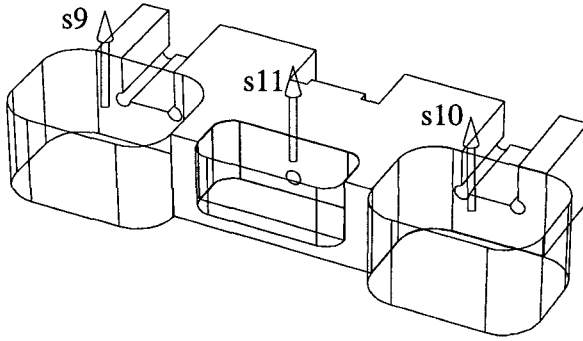


Figure 13: Features of  $P_0^1$  that are different from the features of  $P_0$ .

Use the following procedure (which is described in Section 4.2.1) to find the machining sequence for the features in  $G$  that requires the least setup time:

Let  $T = \text{FIND-BEST-SETUP-TIME}(G, C, \infty, 0)$ .

If  $T < \text{BestTime}$ , then

Return  $G$

5. **Recursion Step:** Pick a feature  $g$  from  $\mathcal{G}$ , and do the following:

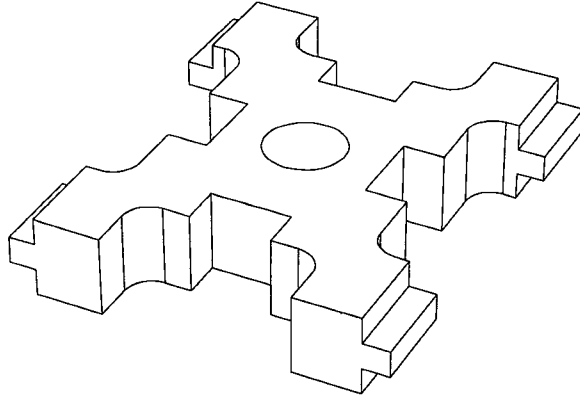
(a)  $\mathcal{R} = \text{GENERATE-REDESIGN-SUGGESTIONS}(L, P, S, \mathcal{G} - g, G, \text{BestTime}, C') \cup \text{GENERATE-REDESIGN-SUGGESTIONS}(L, P, S, \mathcal{G} - g, G \cup g, \text{BestTime}, C')$

(b) Return  $\mathcal{R}$

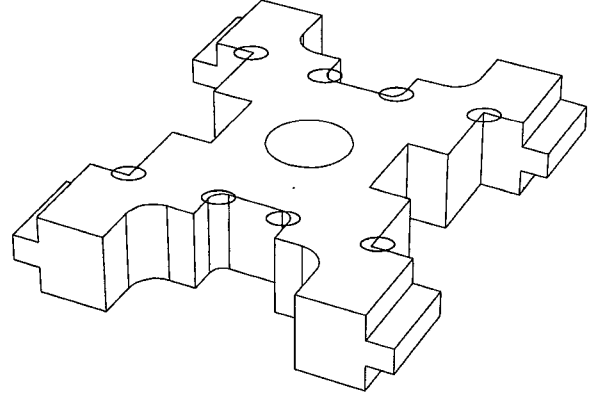
As an example, suppose we apply this algorithm to the features in the set  $\mathcal{F}'$  computed in Section 5. Then we get several RAFSs. For example, Figure 12 shows the parts  $P_0^1$  and  $P_0^2$ . These parts, which are modified versions of  $P_0$ , each can be machined in two setups, so with  $t_s = 2$  minutes the total setup time will be 4 minutes, instead of 8 minutes for the original part. Figure 13 shows which features of  $P_0^1$  are different from those of  $P_0$ .

## 6.1 Other Examples

The methodology can modify different types of part designs to reduce setup time. In Figure 14(a) another example part  $P_1$  is shown. Figure 14(b) shows a modified version of the part. The number of setups required to machine the part  $P_1$  is at least 6, the number of setups required to machine the modified



(a) : Example part  $P_1$



(b) : A Modified Version of the Part  $P_1$  with the Modifications Highlighted.

Figure 14: Example part  $P_1$  and a Redesign Suggestion.

version is 3. Although the level of modification is apparently minor, this kind of modification cannot be generated by modifying the part setup by setup. The count of the setups is effected by the part holding analysis. Without considering work-holding constraints, the count of number of setups will be less than the minimum required.

Another part and a modification suggested by the methodology is shown in Figures 15 (a) and (b). The original part needs at least five setups to machine it, the redesigned part can be machined in two setups.

It should be noted that although the operators we define and use to create feature modifications do not attempt to modify part from the work-holding point of view, the combination of those modifications may also result in improving on the number of setups from work-holding characteristics.

## 7 Discussion and Conclusions

Redesigning a product usually consists of two steps: (1) identifying “redesign clues” (information about what attributes of the design need improvement and why), and modifying these design attributes in order to synthesize an improved design. Existing approaches to this task can be classified as *direct* and *indirect* approaches, as described below.

In *direct* systems [17, 20], rules are used to identify infeasible design attributes from direct inspection of the design description. These infeasible design attributes are then modified using predefined rules to create improved designs. Due to interactions among machining operations, it can be very difficult to determine the manufacturability of a design directly from the design description—and thus the applicability of direct systems is rather limited.

*Indirect* systems [14, 13, 16] proceed by generating a detailed manufacturing plan, and modifying various portions of the plan in order to reduce its cost. Once this has been done designs that correspond to these modified plans are presented to the user as possible redesigns. Although these systems have wider applicability than direct systems, they have several limitations:

1. There may be many possible alternative plans for manufacturing the product, and it is not clear which ones to use as a basis for generating redesign suggestions. Selecting the most promising plans for the initial design may not necessarily produce the best redesign suggestions.
2. If the initial design is not manufacturable, then there will be no plan for the design, and thus no

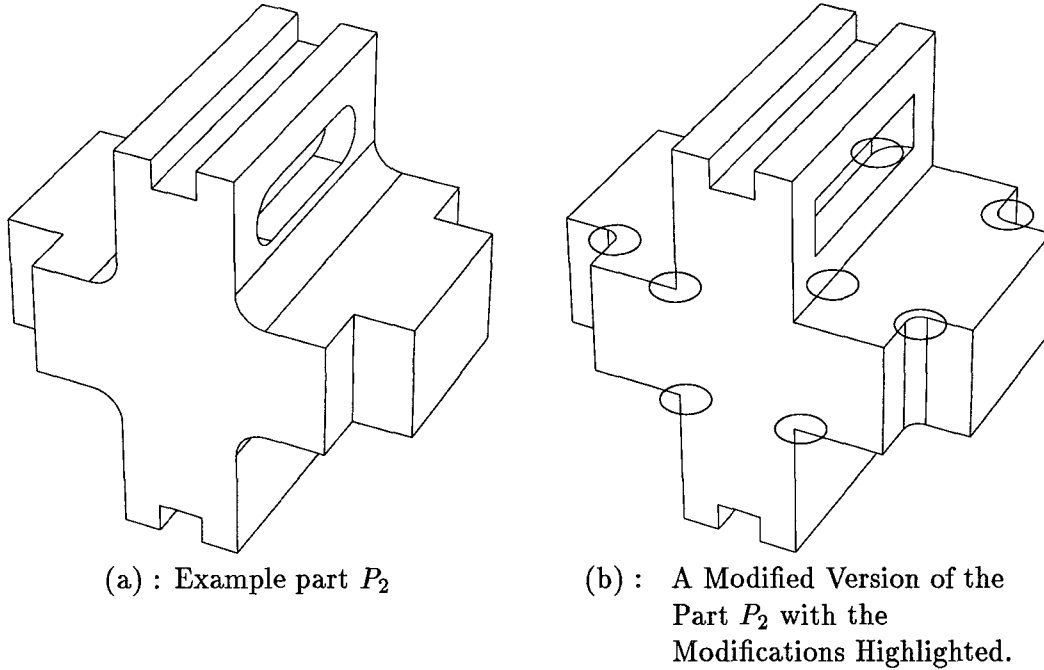


Figure 15: Another Example  $P_2$  and a Redesign Suggestion.

clear way to generate redesign suggestions.

3. Since most existing indirect systems do not take into account the design's functionality, this makes it difficult to ensure that the proposed changes will not violate functionality requirements.

Because of the above limitations, we believe that neither the direct nor the indirect approaches are sufficient by themselves. Thus, our approach incorporates aspects of both the direct and indirect approaches. It uses direct access to the design description so that it can adequately consider the functionality of the design, and it generates setup plans so that it can adequately consider the manufacturability of the design. Some of the other characteristics of our approach are as follows:

1. To represent and analyze the design, we make use of volumetric features that correspond directly to machining operations. These features provide access to some of the geometric information about the design, and also give information about the various alternative ways in which the design might be machined.
2. We generate a set of possible design modifications by modifying some or all of the machining features. In advance of this, we ask the user to assign geometric constraints arising from the intended functionality of the design—and we use these constraints to limit what kinds of possible modifications will be made to the features.
3. Rather than looking at each modified machining feature individually in order to decide whether it improves the part's manufacturability, we generate and consider a number of alternative designs that contain various combinations of modified and unmodified features. In this manner, we can recognize situations in which a combination of modifications can improve the manufacturability of a design even though each individual modification would not improve the manufacturability if it were made by itself.



4. We attempt to generate multiple design alternatives whenever possible, so that the designer can use other kinds of analysis on those redesign suggestions before accepting one as the alternative.

The implementation of the system is not yet complete. For future work, we intend to finish implementation and testing of the system, and we hope to extend its scope in the following ways:

- We are interested in improving our scheme for representing geometric constraints on the design, so that it will better reflect the kinds of restrictions that the designer might want to place on how the design can be changed.
- We are interested in considering geometric and dimensional tolerances of the part while creating local modifications and while generating redesign suggestions.
- For fixturbility analysis we considered flat jaw vise to be the only type of work-holding device. In the future we intend to consider other work-holding devices as well. Work on this is already underway.
- We would like to incorporate manufacturing cost factors other than setups as criterion for generating redesign suggestions.

## References

- [1] G. Boothroyd and P. Dewhurst. *Design for Assembly -A Designer's Handbook*. Department of Mechanical Engineering, University of Massachusetts at Amherst, 1983.
- [2] James G. Bralla, editor. *Handbook of Product Design for Manufacturing*. McGraw Hill Book Company, New York, 1986.
- [3] B. Chandrasekharan, Ashok K. Goel, and Yumi Iwasaki. Functional representation as design rationale. *Computer*, 26(1):48–56, 1993.
- [4] Tien-Chien Chang. *Expert Process Planning for Manufacturing*. Addison-Wesley Publishing Co., Reading, Massachusetts, 1990.
- [5] Diganta Das, Satyandra K. Gupta, and Dana S. Nau. Reducing setup cost by automated generation of redesign suggestions. In Kosuke Ishii, editor, *Proceedings of the ASME International Computers in Engineering Conference*, volume 1, pages 159–170, Minneapolis, MN, 1994. ASME.
- [6] Anthony de Sam Lazaro, David T. Engquist, and Dean B. Edwards. An intelligent design for manufacturability system for sheet-metal parts. *Concurrent Engineering: Research and Applications*, 1(2):117–123, 1993.
- [7] H. A. ElMaraghy, K. F. Zhang, and H. Chu. A function-oriented modeler prototype. In P. J. Guichelaar, editor, *Design for Manufacturability, ASME Winter Annual Meeting*, pages 57–62. ASME, 1993.
- [8] Jin-Kang Gui and Martti Mäntylä. Functional understanding of assembly modelling. *Computer Aided Design*, 26(6):435–451, 1994.
- [9] S. Gupta, W. Regli, and D. Nau. Manufacturing feature instances: Which ones to recognize? In *ACM Solid Modeling Conference*, 1995. To Appear.

- [10] S. K. Gupta, T. R. Kramer, D. S. Nau, W. C. Regli, and G. Zhang. Building MRSEV models for CAM applications. *Advances in Engineering Software*, special issue on *Feature-Based Design and Manufacturing*, 1994. To appear. Also available as Institute for Systems Research Technical Report TR 93-84.
- [11] S. K. Gupta and D. S. Nau. A systematic approach for analyzing the manufacturability of machined parts. *Computer Aided Design*, 1994. To appear. Also available as Institute for Systems Research Technical Report TR 93-76.
- [12] Satyandra K. Gupta. *Automated Manufacturability Analysis of Machined Parts*. PhD thesis, University of Maryland, 1994.
- [13] C. C. Hayes, S. Desa, and P. K. Wright. Using process planning knowledge to make design suggestions concurrently. In N. H. Chao and S. C. Y. Lu, editors, *Concurrent Product and Process Design, ASME Winter Annual Meeting*, pages 87–92. ASME, 1989.
- [14] C. C. Hayes and H. C. Sun. Plan-based generation of design suggestions. In R. Gadh, editor, *Concurrent Product Design, ASME Winter Annual Meeting*, pages 59–69, New York, 1994. Design Engineering Division, ASME.
- [15] Mark R. Henderson and LeRoy E. Taylor. A meta-model for mechanical products based upon the mechanical design process. *Research in Engineering Design*, 5(4):140–160, 1993.
- [16] Wynne Hsu, C. S. George Lee, and S. F. Su. Feedback approach to design for assembly by evaluation of assembly plan. *Computer Aided Design*, 25(7):395–410, July 1993.
- [17] M. Jakiela and P. Papalambros. Design and implementation of a prototype intelligent CAD system. *ASME Journal of Mechanisms, Transmission, and Automation in Design*, 111(2), June 1989.
- [18] M. Kanumury and T. C. Chang. Process planning in an automated manufacturing environment. *Journal of Manufacturing Systems*, 10(1):67–78, 1991.
- [19] Mark Klein. Capturing design rationale in concurrent engineering teams. *Computer*, 26(1):39–47, 1993.
- [20] Prashant V. Mahajan, Corrado Poli, David W. Rosen, and Michael J. Wozny. Design for stamping: A feature-based approach. In P. J. Guichelaar, editor, *Design for Manufacturability, ASME Winter Annual Meeting*, pages 29–49. ASME, 1993.
- [21] M. Mäntylä, J. Opas, and J. Puhakka. Generative process planning of prismatic parts by feature relaxation. In B. Ravani, editor, *Advances in Design Automation, ASME Winter Annual Meeting*, pages 49–60. ASME, 1989.
- [22] G. Pahl and W. Beitz. *Engineering Design*. Design Council, London, 1984.
- [23] William C. Regli, Satyandra K. Gupta, and Dana S. Nau. Extracting alternative machining features: An algorithmic approach. *Research in Engineering Design*, 1995. To appear.
- [24] David Robertson, Karl T. Ulrich, and Marc Filerman. A structured design methodology and metadesigner: A system shell concept for computer aided creative design. In B. Ravani, editor, *Proceedings of the Advances in Design Automation Conference*, pages 308–313. ASME, 1989.

- [25] Reinhard Schiebeler and Klaus Ehrlenspiel. Reek: A knowledge-based system as an integrated design-assistant. In P. J. Guichelaar, editor, *Design for Manufacturability, ASME Winter Annual Meeting*, pages 69–74. ASME, 1993.
- [26] Michael Schulte, Christian Weber, and Rainer Stark. Functional features for design in mechanical engineering. *Computers in Industry*, 23(1):15–24, 1993.
- [27] LeRoy E. Taylor. *Meta-Physical Product Modelling*. PhD thesis, Arizona State University, 1993.
- [28] R. V. Welch and J. R. Dixon. Extending the iterative redesign model to configuration design: Sheet metal brackets as an example. In W. H. ElMaraghy, W. P. Seering, and D. G. Ullman, editors, *Proceedings of the Design Theory and Methodology Conference*, pages 81–88. ASME, 1989.
- [29] F. W. Wilson and P. D. Harvey. *Manufacturing Planning and Estimating Handbook*. McGraw Hill Book Company, 1963.

## A Details of the Feature Modification Operators

Here we present detailed descriptions of some feature modification operators.

### A.1 Operators for modifying drilling features

$\mathcal{O}_d^d$  : Creating new drilling feature

Applicability Condition: Drilling features that correspond to blind holes i.e. where the intersection of the conical bottom of the feature and the part is non-null.

The new drilling feature  $f'$  will be a through drilling feature with approach direction  $-\vec{v}(f)$ . After that feature is created another through drilling feature from the same direction as the current feature will also be added to  $\mathcal{F}'$ .

The diameter  $d(f')$  of the new feature will be the same as the diameter of the feature being modified. The datum point and approach plane will be determined in the following manner.

A semi-infinite cylinder  $C$  will be created with diameter  $d(f')$  starting at the approach plane with the axis going through the datum point of the drilling feature  $f$ . It will extend infinitely in the direction  $\vec{v}(f')$ . This cylinder  $C$  will be bounded by a plane perpendicular to  $\vec{v}(f')$  and tangential to the stock. The datum point of  $f'$  will lie on the axis of  $C$  on this plane. The length  $l(f')$  of the new feature will be the distance between the datum points of the features  $f$  and  $f'$  with the height of the conical bottom added to it.

After this modification another through drilling feature with the same diameter and datum point of the old feature  $f$  and the length of the new drilling feature  $f'$  will be created.

It should be noted that the new feature might violate the depth to diameter ratio for a drilling feature. If the new feature becomes part of a redesign suggestion, the designer will have the option of changing the diameter at that point.

$\mathcal{O}_d^e$  : Creating new end milling feature

Applicability Condition: All drilling features in part  $P$ .

One end milling feature  $f'$  is created from each approach direction in  $D$  (defined in Section 5.2) which are perpendicular to the approach direction  $\vec{v}(f)$ . The width of the feature  $w(f')$  will be the diameter of the drilling feature  $f$ . The length of the feature  $d(f')$  will be equal to  $l(f) + 0.5(w(f'))$ . We get the

complete feature by first defining six planes to enclose a rectangular volume and then blending the edges parallel to the approach direction  $\vec{v}(f')$  to half of the width of the feature. We define those six planes below.

A plane  $p_1$  perpendicular to  $\vec{v}(f')$  and tangential to the drilling feature such that the axis of the drilling feature will lie in the direction  $\vec{v}(f')$  from the plane. A plane  $p_2$  perpendicular to  $\vec{v}(f')$  and tangential to the stock lying in the direction  $\vec{v}(f')$  from  $p_1$ . This will also be the approach plane for the new feature  $f'$ . Two planes  $p_3$  and  $p_4$  perpendicular to  $p_1$  and  $p_2$  and tangential to the drilling feature. Plane  $p_5$  perpendicular to planes  $p_1$  through  $p_4$  and tangential to the tip of the conical bottom of the drilling feature. Plane  $p_6$  perpendicular to planes  $p_1$  through  $p_4$  and parallel to  $p_5$  at a distance  $d(f')$  from  $p_5$  in the direction of  $\vec{v}(f)$ .

The feature such obtained might have to be translated at a direction  $\pm(\vec{v}(f) \times \vec{v}(f'))$  to avoid intersection with faces in the original part.

$\mathcal{O}_d^s$  : Creating new side milling feature

Applicability Condition: All through drilling features in part  $P$ .

Two side milling features are created for each approach direction in  $D$  which are perpendicular to  $\vec{v}(f)$ . The depth of the features  $l(f')$  will be the same as the diameter of the drilling feature. The length and width of the features will be determined in the process of generating the features. New features are generated in a manner similar to  $(\mathcal{O}_d^e)$ . We first define six planes to contain a rectangular solid, then we blend appropriate edges to maximum possible blending radius and then trim the feature.

The first two such planes would be  $p_1$  and  $p_2$  which are perpendicular to  $\vec{v}(f')$  and tangential to the drilling feature. Next we find auxiliary planes  $p_{a1}$  and  $p_{a2}$  parallel to  $\vec{v}(f')$  and tangential to the stock. For each of these planes we get one new side milling feature. So we find the four other planes to get a rectangular solid for each plane in  $p_a$  ( $p_a = p_{a1} \cup p_{a2}$ ).

Plane  $p_3$  perpendicular to  $p_1$  and  $p_2$  and tangential to the drilling feature. Plane  $p_4$  parallel to  $p_3$  at a distance  $2 \times s$  (where  $s$  is the distance between  $p_4$  and  $p_a$ ) such that the drilling feature lies between  $p_3$  and  $p_4$ . Planes  $p_5$  and  $p_6$  will be perpendicular to  $\vec{v}(f)$ . These two planes will be located on two sides of the stock at a distance  $2 \times s$  from a plane tangent to the stock.

Once we get these six planes and the rectangular solid enclosed by it we blend the two edges on the plane  $p_3$  and parallel to  $\vec{v}(f')$  to a radius equal to  $2 \times s$ . After that we trim the solid by the auxiliary plane  $p_a$  to get the side milling feature  $f'$ .

The feature such obtained might have to be translated at a direction  $\pm(\vec{v}(f'))$  to avoid intersection with faces in the original part.

## A.2 Operators for modifying end milling features

The end milling features can be converted to other end milling features with a different approach direction or to other milling features, face milling and side milling. End milling features will not be converted to drilling features.

$\mathcal{O}_e^e$  : Creating new end milling features

Applicability Condition: All end milling features in the original part  $P$  which can be represented by parameters described in Section 5.2.

First a rectangular solid  $A$  of minimum size is created which encloses the end milling feature  $f$ . We will create four features with approach directions opposite to the face normals of the four faces in  $A$  which are parallel to  $\vec{v}(f)$ .

For each face  $p$  which is parallel  $\vec{v}(f)$  that is extruded infinitely in the direction opposite to its face normal to get a solid  $I$ .  $I$  is trimmed by a face parallel to  $p$  and tangential to the stock. The edges in solid  $I$  which are parallel to  $\vec{v}(f')$  are blended and the relevant edges offsetted to create an end milling feature  $f'$ . The offsetting is done in such a way so as not to eliminate faces in the part. Only face which can be eliminated is the one parallel to  $p$ .

## B Details of the different tests performed while building admissible feature sets

In the procedures `ANALYZE-DESIGN` and `GENERATE-REDESIGN-SUGGESTIONS` we perform four tests on the feature set  $G$  being built to check if the feature set will result in an AFS or not. These tests are, pruning, redundancy, feasibility and goal. The pruning and redundancy tests are explained in the procedures. The other three tests are explained below.

**Feasibility Test:** A feature set will be considered non-feasible from two considerations. It is to be noted that any feature added to the set will only result in more material being removed from the stock.

1. If the solid generated by subtracting the features in  $G$  from the stock results is non-manifold or disjoint, then that set will not result in an AFS. So we check for that in first step.
2. If the solid created by subtracting the features in  $G$  from the stock violates some geometric constraints which cannot be satisfied by adding other features to the set. Some of this types of constraint violations are listed below. Note that not all constraint violation or satisfaction can be checked before a complete AFS is built. We check for only those constraint satisfactions (or violations) which can be checked with a partial AFS. Some examples of such violations are listed below. Let  $V$  be the solid generated by subtracting the partial AFS from the stock  $S$ .
  - (a) The area of contact of a constrained face in a constraint volume with the boundary of  $V$  is less than the minimum permissible
  - (b)  $V$  contains a blended edge, the blending radius of which is more than the maximum permissible

**Goal Test:** This test is performed to check if the feature set  $G$  is an AFS or not. This is a two step test. In the first step we will create a solid  $W$  by subtracting the features of the AFS being built from the stock  $S$  and test the existence of the constraint volumes specified. This test will be done after the blending of the blendable edges on the constraint volumes. If this test fails then the feature set is not an AFS. If the test is successful, then we will test for the constraint satisfaction on the relevant faces and edges of the constraint volumes. If that test is successful, then  $G$  is an AFS.

## C Details of the procedure `HOLDING-ANALYSIS`

### C.1 Probing methodology

Whenever we have a possible workpiece to investigate for viability of a setup, we need to find whether there exists geometric features on the workpiece which can be used to establish a datum on the part for CNC machining. If that is not possible we will discard any setup sequence which will require us to machine that workpiece. Kanumury *et al.* gave details about the need and procedure of probing in their article [18].

At Step 5c of the procedure `FIND-BEST-SETUP-TIME` described in Section 4.2.1 we check the workpiece for feasibility of probing it for locating on a machine table. The procedure `WORKPIECE-PROBE` returns

true if it is feasible to probe the workpiece and returns false otherwise. The feasibility is determined by checking for the existence of already machined faces or stock faces which are accessible from the top, in the workpiece that allow establishing a datum point for machining the features. We assume that the following combinations of faces will allow establishment of datum.

1. Three mutually perpendicular planar faces, one of which is perpendicular to the approach direction
2. One face perpendicular to the approach direction, a second face perpendicular to that face and a cylindrical face with axis perpendicular to the approach direction
3. One face perpendicular to the approach direction and two cylindrical faces with axis perpendicular to the approach direction

## C.2 Work-Holding Analysis

We assume that only flat vise jaws are to be used for holding the workpiece during machining. A vise is a pair of rectangular jaws. The workpiece needs to be secured by putting two vise jaws against two parallel faces on the workpiece. For properly holding the workpiece the minimum projected area of those two parallel faces between the jaws have to be more than a specific minimum area  $A_t$ .

In Section C.2.1, we describe the details of how the work-holding analysis works. We do not suggest the exact setup locations for different setups. We only determine which features can be machined in one setup using vise jaws as work-holding device.

### C.2.1 Analysis for Vise Clamping

This section describes the HOLDING-ANALYSIS procedure that is used in Step 5(c)i of FIND-BEST-SETUP-TIME described in Section 4.2.1. For this analysis the workpiece is oriented to have the features in set  $H$  facing upwards with their approach direction  $\vec{v}$  perpendicular to the machine table. It is also assumed that the workpiece is kept at a fixed position on the machine table and the vise jaws are moved around to hold the workpiece.

The procedure HOLDING-ANALYSIS takes as argument the current workpiece  $W$ , the set of features under consideration  $H$ , the approach direction  $\vec{v}$  for the features in  $H$  and the set of precedence constraints  $C$  among the features. First it finds out the face pairs which can be used to hold the workpiece which are parallel to each other, is accessible to the vise jaws and has a minimum projection area on each other  $A_t$ . This value of this threshold minimum area  $A_t$  depends on the cutting force required to machine the part. The procedure calculates the set of distances ( $\Gamma$ ) of the features which might possibly intersect with the vise jaws, from a plane  $\lambda$  tangent to the bottom of the workpiece from the bottom of the part (for example, Figure 17 and 18 show how this calculation is performed for the workpiece shown in Figure 16).

After calculating this set  $\Gamma$ <sup>11</sup> the procedure HOLDING-ANALYSIS calls the procedure FIND-FEATURES-IN-SETUP which takes as argument the approach direction  $\vec{v}$ , the workpiece  $W$ , the bottom plane  $\lambda$  of  $W$ , the set of features  $H$  under consideration and the sets of pairs of holding faces  $Z$  and the set  $\Gamma$ . It also takes as argument the set of precedence constraints among the features in  $H, C$ . It returns the set  $\mathcal{K}$ , which contains subsets of  $H$  which can be machined in one setup.

**procedure** HOLDING-ANALYSIS( $W, \vec{v}, H, C$ )

1. Let  $R$  be the set of all planar faces in  $W$  such that for each  $r \in R$ ,  $r$  is accessible to the vise jaw from the direction opposite to its face normal i.e.

---

<sup>11</sup>The set of vertical distances  $\Gamma$  are the vertical locations at which the tops of the vise jaws will be aligned to test for interference with workpiece and the features. Later we also find similar lateral locations to align the jaws for testing.

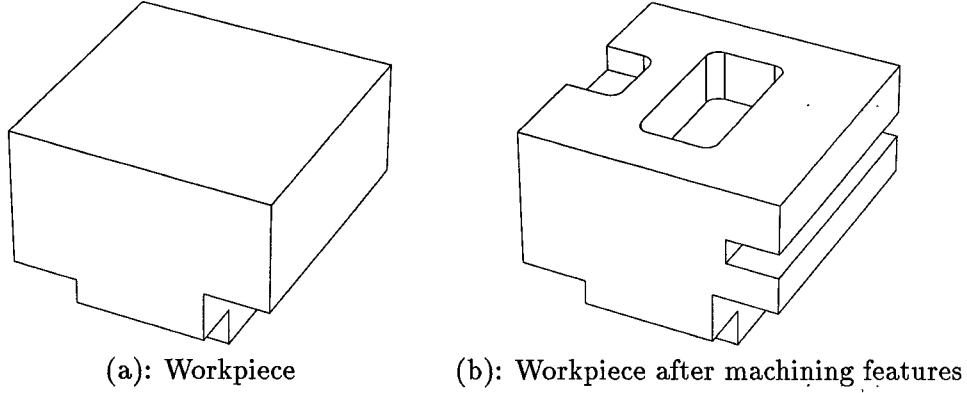


Figure 16: Workpiece for Holding Analysis

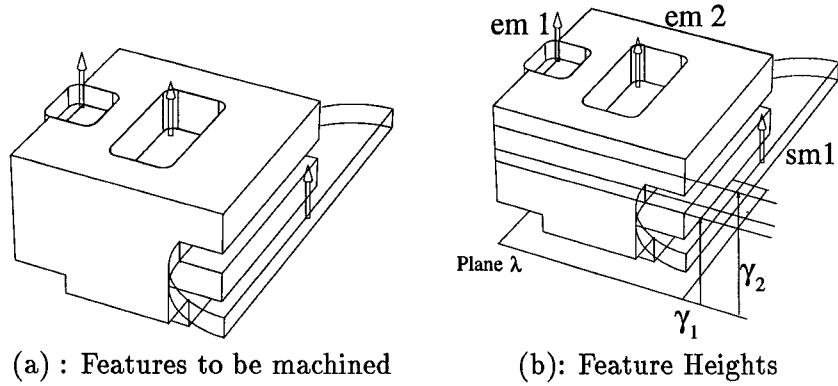


Figure 17: Diagram for Holding Analysis - I

- (a)  $L \cap^* W = \emptyset$ , where  $L$  is the swept volume produced by sweeping  $r$  infinitely in the direction of its face normal
  - (b)  $r$  is accessible from the direction opposite to the approach direction  $\vec{v}$
- (Thus  $R$  is the set of all faces that can potentially be used as holding faces. For example, for the workpiece shown in Figure 16,  $R$  consists of the faces  $a1, a2, a3, b1, b2, c1, c2$  shown in Figure 18).
2. Let  $\lambda$  be the plane touching the workpiece  $W$  which is perpendicular to the approach direction  $\vec{v}$  and tangent to the bottom of the workpiece.  
(For example see Figure 17)
  3. Let  $Z$  be the set of all face pairs  $(p_i, p_j) \in R$  such that
    - (a)  $p_i, p_j$  are parallel
    - (b) The face normals of  $p_i$  and  $p_j$  have opposite direction.
    - (c)  $A \geq A_t$   
 where  $A$  is the area of the projection of  $p_i$  on  $p_j$  and  $A_t$  is the minimum holding area for vise clamping  
 (The set  $Z$  will contain candidate face pairs which can be used for clamping in vise in a stable manner. For example, in the case of the faces shown in Figure 18,  $Z$  would contain the face pairs  $(a1, a2), (b1, b2), (c1, c2)$  but would not contain  $(a1, a3)$ , because the area of projection of  $a1$  on  $a3$  is less than  $A_t$ .)

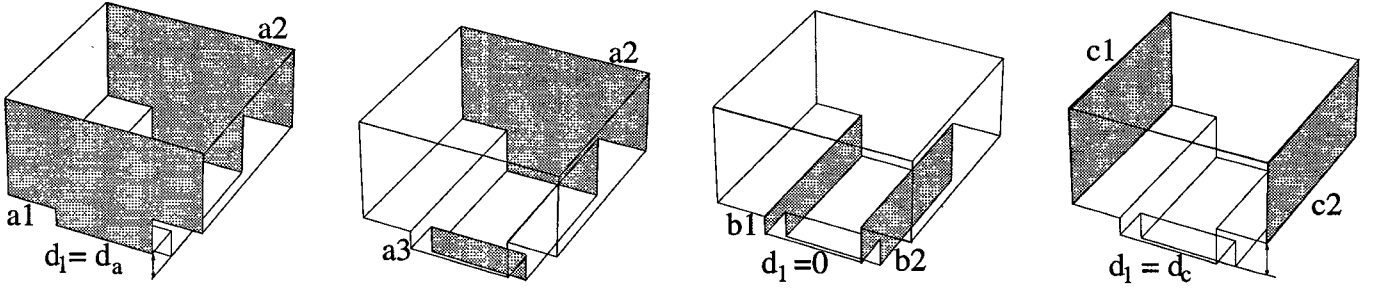


Figure 18: Diagram for holding Analysis -II

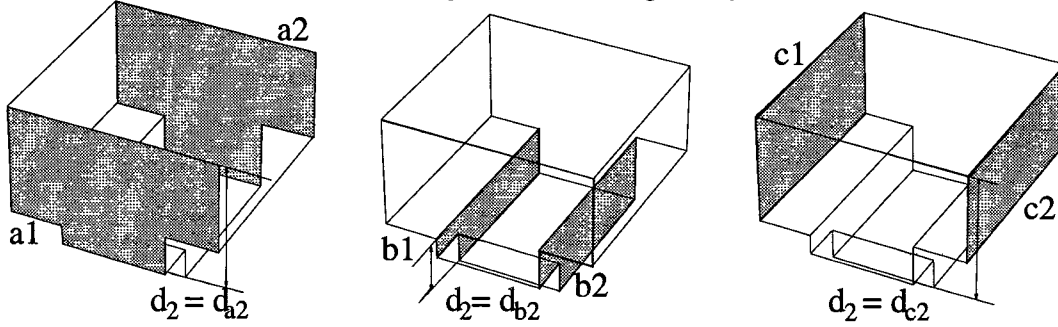


Figure 19: Diagram for Holding Analysis - III

4. For each  $h \in H$  which intersects with any face in the set of face pairs  $Z$ , let  $\gamma_h$  be the minimum distance from  $\lambda$  to the feature  $h$ , measured along the approach direction  $\vec{v}$
5. Let  $\Gamma$  be the set of all the  $\gamma_h$ 's
6.  $\mathcal{K} = \text{FIND-FEATURES-IN-SETUP}(\lambda, \Gamma, \vec{v}, W, H, Z, C)$
7. Return  $\mathcal{K}$

Figure 17(a) shows three features which can be machined in one setup if no fixturing problems exist. Figure 17(b) shows the plane  $\lambda$  and the distances  $\gamma$  from the that plane to the features. Note that the distance to feature em 2 from the bottom plane is not computed, because in this orientation that feature is not going to interfere with the vise jaws.

HOLDING-ANALYSIS is *sound*, in the sense that it will only find the face pairs which will allow proper work holding. However, it will fail to identify some possible faces and face pairs which could have been used for the clamping purpose. These cases are:

1. Some faces which are partially accessible can be used as holding face, but this procedure will reject those as holding faces.
2. Some times different faces on same plane can be used together for holding the workpiece, but this algorithm does not consider that.

For the purpose of analyzing how to put features in one setup we assume the vise to be a pair of two identical rectangular solids  $J_1$  and  $J_2$ . The length of these jaws are more than the longest linear dimension of the workpiece  $W$  measured along the face normals of all the faces  $r \in R$ . The height of the vise jaws is more than the longest linear dimension of the workpiece  $W$  measured in a direction parallel to  $\vec{v}$  along all  $r \in R$ . We assume that the vise jaw opening is sufficient to cover the distance between any pair of faces in



$Z$ . The two vise jaws will be aligned with the faces of the set of face pairs  $Z$  along the length of the jaws at different locations for finding which features can be machined in one setup.

The conditions for a subset of features in  $H$  to be machinable in one setup are the following:

1. The vise jaws will not intersect with the features removal or accessibility volume
2. All the precedence(s) of the features in that subset also have to belong to that subset.

**procedure** FIND-FEATURES-IN-SETUP( $\lambda, \Gamma, \vec{v}, W, H, Z, C$ )

1. (Below, we compute sets  $K_L$  and  $K_R$  of features which can be machined in one setup using the face pair for holding the workpiece with the vise jaw  $J_2$  respectively to the left and right of the face  $p_j$ .) Initially, set  $K_L = K_R = \emptyset$
2. For each face pair  $(p_i, p_j) \in Z$  (in increasing order of the total area of intersection with features in  $H$ , in case of more than one face pair having the same area of intersection with features, the one with higher area will be selected), do the following:
  - (a) Let  $d_1$  and  $d_2$  be the shortest and longest distance from  $\lambda$  to any point in  $A$ , where  $A$  is the area of projection of  $p_i$  on  $p_j$  (see Figure 18 and 19)
  - (b) Let  $\Gamma_s = \{d_2\} \cup \{d \in \Gamma : d_1 < d < d_2\}$   
( $\Gamma_s$  contains the possible vertical locations where the face pair might potentially be aligned with the top of the vise jaws.)
  - (c) (Below, we compute sets  $K_L^*$  and  $K_R^*$  of features which can be machined in one setup using the face pair for holding the workpiece with the vise jaw  $J_2$  respectively to the left and right of the face  $p_j$  at the vertical position  $\gamma$  of the workpiece.) Initially, set  $K_L^* = K_R^* = \emptyset$
  - (d) For each  $\gamma \in \Gamma_s$  do the following in the order of increasing value of  $\gamma$  (the higher the value of  $\gamma$ , the higher the portion of the workpiece located inside the vise jaws):
    - i. If the area of the projection  $A$  of face  $p_i$  on  $p_j$  below the height  $\gamma$  is less than  $A_t$ , then exit, because there is not enough holding area to hold the workpiece securely in the vise. Otherwise, do the following:
    - ii. If  $K_L^* = \emptyset$  then
 
$$K_L^* = \text{LEFT-HOLDING-ANALYSIS}(W, H, A, A_t, C, \gamma)$$

$$K_L = K_L \cup K_L^*$$
 If  $K_L^* = H$ , then then set  $\mathcal{K} = H$  and go to Step 4, because we found that all the features can be machined in one setup and we need not search any more.
    - If  $K_R^* = \emptyset$  then
 
$$K_R^* = \text{RIGHT-HOLDING-ANALYSIS}(W, H, A, A_t, C, \gamma)$$

$$K_R = K_R \cup K_R^*$$
 If  $K_R^* = H$ , then set  $\mathcal{K} = H$  and go to Step 4, because we found that all the features can be machined in one setup and we need not search any more.
    - (If  $K_L^*$  or  $K_R^*$  is non-empty then we need not call LEFT-HOLDING-ANALYSIS or RIGHT-HOLDING-ANALYSIS, respectively, because no more features will be accessible than before.)
3.  $\mathcal{K} = K_L \cup K_R$
4. Return  $\mathcal{K}$

As an example, for the workpiece shown in Figure 16, the input parameter  $Z$  to FIND-FEATURES-IN-SETUP consists of three face pairs (a1,a2), (b1,b2) & (c1,c2) that can possibly be used for aligning the vise jaws. Figures 17,18 and 19 show the parameters used and calculated by FIND-FEATURES-IN-SETUP. As the face pair (a1,a2) does not intersect with any features and has the maximum area that will be considered first by FIND-FEATURES-IN-SETUP. For this face pair,  $\Gamma_s = \{\gamma_1, \gamma_2, d_{a2}\}$ . For face pair (c1,c2),  $\Gamma_s = \{\gamma_1, \gamma_2, d_{c2}\}$ . For the face pair (b1,b2),  $\Gamma_s$  will consist of only  $d_{b2}$ , as all the values in  $\Gamma$  are more than  $d_{b2}$ . FIND-FEATURES-IN-SETUP will not examine all of these face pairs, because all three features are accessible using the face pair (a1,a2).

The procedures LEFT-HOLDING-ANALYSIS and RIGHT-HOLDING-ANALYSIS analyze the face pair, their projection area on each other and the vise jaws to find the features accessible in different vise positions with respect to the workpiece. In both of these procedures we locate the workpiece with respect to the vise at different vertical positions and at the extreme lateral positions, (left and right), such that the minimum area of contact of the vise jaws with the faces being considered is the minimum threshold area  $A_t$ . These procedures compute the feature sets which can be machined in those locations. In these two procedures, we assume that appropriate spacer bars are available to position the workpiece at any desired height.

**procedure** LEFT-HOLDING-ANALYSIS( $W, H, A, A_t, C, \gamma$ )

1. Let  $L$  be a vertical line on the area  $A$ , such that the area of the patch on  $A$  below the height  $\gamma$  and left of  $L$  is  $A_t$ <sup>12</sup>
2. Let  $J_2$  be a vise jaw (as described in the text) such that  $J_2$ 's inside face (the one facing the workpiece) lies on the face  $p_j$ ,  $J_2$ 's top edge is at a height  $\gamma$  on the face  $p_j$ , and  $J_2$ 's rightmost vertical inside edge is collinear with  $L$ .
3. Let  $J_1$  be a vise jaw whose inside face is at the corresponding location on  $p_i$ .
4. If the jaws  $J_1$  and  $J_2$  intersect with the workpiece  $W$ , then return  $\emptyset$  because the workpiece cannot be properly held at that position
5. Otherwise, find the set of features  $K^* \subseteq H$  which are accessible for machining at that location  
If  $K^*$  contains any feature  $k'$  such that the precedence of  $k'$  is not in  $K^*$ , then  $K^* = K^* - k'$
6. Return  $K^*$

RIGHT-HOLDING-ANALYSIS is an identical procedure where we place the vise jaw to the right of the workpiece instead of to the left of the workpiece as done in procedure LEFT-HOLDING-ANALYSIS. We do not suggest the vertical and lateral locations where we locate the workpiece with respect to the vise to be the ideal location for clamping. We choose these locations, because we can estimate the maximum number of features which might be accessible for machining by checking at those locations.

---

<sup>12</sup>Although it is difficult to compute an exact value for  $L$ , a good approximation can be computed reasonably quickly using binary search.