Computer-Aided Design, Vol. 29, No. 10, pp. 687–699, 1997

Printed in Great Britain
0010-4485/97/$17.00+0.00
PII: S0010-4485(97)00014-6

Resolving feature interactions in 3D part editing

Der-Baau Perng†* and Chao-Fan Chang†

Feature-based design approach has been regarded as a promising approach for 3D parts design. However, the critical issues of modifying a part design by 3D features have not yet evoked sufficient discussion. This paper aims to address issues of feature interaction, especially for enclosure and intersection, among the modified feature and the other existing features. An efficient new approach to solve the feature interaction problems encountered in part-editing is proposed. The parts are assumed to consist of subtractive volume-features only. A set of rules that facilitate updating the B-rep data and feature-based representation of the part are devised. Computer simulation examples are given to show that the proposed approach is both feasible and effective. This research contributes to several aspects of feature-based design research, especially to the area of providing simple feature-based commands. © 1997 Elsevier Science Ltd

Keywords: design by features, feature interaction

INTRODUCTION

The feature-based approach, using high-level features as entities and communication roles, for part design has been regarded as the key means of linking the design and manufacture of parts[1,4,14]. For example, Chan and Voelcker[2] adopted a feature-based design system to develop their machining process plan language. The parts designed by most feature-based design systems are usually described by a volume-based approach[10] of hybrid B-rep (Boundary Representation)/CSG (Constructive Solid Geometry) representation scheme[19]. In such feature-based design research, the critical issues of modifying a part designed by 3D features have not yet evoked sufficient discussion. Only a few studies[12,17] have discussed the topic of 3D feature-based editing.

In editing a feature-based designed part, we will encounter two problems for the part description:

(1) The B-rep of the modified part will be changed.
(2) The CSG representation (or feature list) of the modified part will be changed.

*Institute of Industrial Engineering, National Chiao Tung University, Hsin chu, Taiwan 30010, R.O.C.; Tel: 886-35-712121 ext. 57319; Fax: 886-35-722392.
†Institute of Industrial Engineering, National Chiao Tung University, Hsin chu, Taiwan 30010, R.O.C.
Paper Received: 20 November 1995. Revised: 27 September 1996

Because the modified feature may enclose or intersect with other existing features of the part, such feature enclosure and intersection cases are systematically analyzed first in this paper. Several feature-interaction properties are deduced. Then an efficient approach is proposed to solve such feature-based editing problems. The framework of the proposed approach is shown in *Figure 1*. We update and display the whole B-rep of the designed part in an efficient way, and process the CSG-tree or feature list of the designed part to eliminate any redundant features. Consequently, designers can easily construct and modify parts by specifying the parameters of the volume-features. Moreover, the resulting part will be consistent with the designer's intent.

This paper is organized as follows. We first review previous works on feature-based editing and feature interaction solving methods. Then, we introduce some assumptions and important properties before solving the feature interaction problems. Third, we describe the derivation process about the two-stage feature interaction solving approach. Fourth, we use examples to demonstrate the proposed feature interaction solving approach. Finally, we give summaries and suggestions.

REVIEW OF PREVIOUS WORKS

Several issues on feature-based editing have been discussed by Rossignac[12]. He adopted the concept of active zones[11] and a spatial decomposition scheme, the SGC[13], to improve the performance of updating the B-rep of a part model when a volume-feature is modified. He defined the active zone of a primitive in a solid as the region where changes to a primitive affect the solid[10]. The purpose of the active zone is to reduce redundant computation while re-executing the Boolean operations in a CSG tree. However, the essence of Rossignac's approach is that he used the primitives and the related Boolean operations to re-evaluate the boundary of a modified part.

Su *et al.*[17] proposed a three-phase method for feature interaction resolution based upon the Extended CSG Tree of Features (ECTOF) scheme[6] for feature representation. In each modification action, their system will perform interference checking and rearrange the ECTOF to have the resulting part be consistent with the user's intent. To re-evaluate the B-rep of the modified part, they must re-execute all the Boolean operations about the nodes in the ECTOF. They did not put further effort into improving the re-evaluation performance.

Suh and Ahulwalia[16] presented an approach to handle the feature interaction problem in incremental feature generation. They tried to redefine or modify the existing

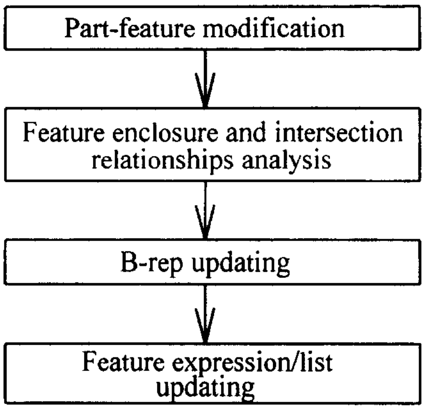


Figure 1 The proposed framework of 3D feature-based editing approach

feature automatically while a new feature is added. However, they did not discuss the re-evaluation of the part's B-rep and only considered two features in each interaction case.

The recent commercial CAD systems with parametric form features have created some confusion about feature modelling[5]. These systems, including Pro/ENGINEER, Bravo, CADDS 5,I-DEAS, Unigraphics, etc., are not true volume-feature-based modelling systems. These systems used surface-based features as design entities. This is inconvenient for designers to perform the feature editing operations. Features in these systems are merely viewed as macros that facilitate the creation, parameterization and placement of specific geometric forms within a solid modeller. In contrast to a feature's ability to capture high-level information, such as, forms, functions, designer intent, material properties, technological parameters, and manufacturing precision, the features in these CAD systems contain only the description of their geometric form[3]. Moreover, the primitives used in a solid modeller are suitable for pure geometric calculation rather than high-level information queries; most of the interpretations about designed parts must be handled by human beings before the part model is processed by the downstream applications.

For the two feature-based editing problems, the previous approaches focused only on either B-rep or CSG updating. Instead of re-executing the Boolean operations about the primitives in the CSG tree while re-evaluating the boundary, we use the original evaluated B-rep and the features that may affect the design result to evaluate the derived B-rep. Without complex efforts on constructing the active zone and SGC, we examine the feature redundancy of a part description by spatial enclosure checking. Besides, we allow more than two features in each feature interaction case. Therefore, a simple and easy operated volume-feature interaction solving approach for both B-rep and CSG updating problems is proposed in this research.

RELATED PROPERTIES OF THE TWO-STAGE FEATURE INTERACTION SOLVING APPROACH

In this section, the feature model and assumptions of a designed part are first introduced. To analyze the enclosure-and-intersection relationships among features, five related properties which influence the designed part, are proposed. These feature interaction properties will be used for deriving the two-stage feature interaction solving approach in the next section.

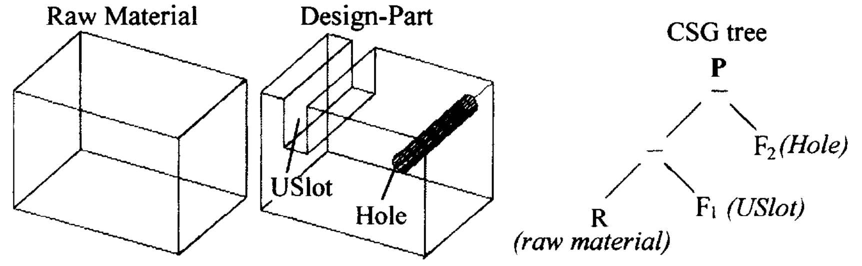
The feature model and assumptions of a designed part

Features discussed in this research are the volumes to be removed by machining operations[9,16]. Each feature of a part can be treated as a subpart than can be subtracted from the raw material of the part. The domain of subtractive volume-features in this research includes: arch, fillet, hole, step, Tslot, Uslot, Vslot, and wedge. Parts can be constructed and modified by specifying the parameters of the volume-features.

A feature-based part can be described by either a feature expression or a feature list. The feature expression of a part represents an expression of a CSG form consisting of a raw material, feature operands, and Boolean operators. The feature list, on the other hand, represents a link list of features of the part model. For example, *Figure 2* demonstrates a feature-based part and the mapping relationship of the terminology here defined.

Two assumptions about a designed part are given below.

- Only the Boolean operator of "difference"(−)[18] can exist in the final feature expression of a part model. Such a feature expression is actually of a DSG (destructive solid geometry) form[8].



Feature list: F_1(*USlot*), F_2(*Hole*).
R: raw material.
Feature expression: $P=(R-F_1)-F_2$.

Figure 2 A demonstrated design part and corresponding terminology of feature expression, feature list, raw material, Boolean operator and feature operand

- No feature of a part can be enclosed completely by another feature, whether in designing or editing a part.

Feature interaction properties

The feature enclosure and intersection cases between the modified feature and the other existing features of the designed part are the critical issues to be discussed in the 3D feature-based editing problems. The effects of the feature-enclosure relationship on the feature expression are introduced in properties 1, 2, and 3 below. Properties 4 and 5 describe how to deal with the part model when the feature-intersection relationship exists.

Property 1: enclosure property
For a part P expressed as removing features F_1 and F_2 from the raw material R, and the feature F_2 is the new constructed feature

$$P=(R-F_1)-F_2$$

If $F_1 \subset F_2$, then $P=R-F_2$.

Property 2: exchange property
For two parts P and P' expressed as

$$P=(R-F_1)-F_2 \text{ and } P'=(R-F_2)-F_1$$

If $F_1 \not\subset F_2$ and $F_2 \not\subset F_1$, then $P=P'$.

Property 3: redundancy property
For a part P expressed as

$$P=(R-F_1)-F_2$$

Subtract the feature F_1 again from the part model, i.e.

$$P'=((R-F_1)-F_2)-F_1$$

If $F_1 \not\subset F_2$ and $F_2 \not\subset F_1$, then $P'=(R-F_1)-F_2$ and $P'=P$.

Property 4: counteraction property
For a part P with two features F_1 and F_2 expressed as

$$P=(R-F_1)-F_2, \text{ where } F_1 \cap F_2=\varnothing$$

When a new part P' is constructed by uniting the feature F_1 with P

$$P'=((R-F_1)-F_2)+F_1$$

then $P'=R-F_2$.

Property 5: intersection property.
For the Part P expressed as

$$P=(R-F_1)-F_2, \text{ where } F_1 \cap F_2=I,\ I \neq \varnothing$$

When a new part $P'$is constructed by uniting the feature F_1 with P

$$P'=((R-F_1)-F_2)+F_1$$

then $P'=(R-F_2)-I$.

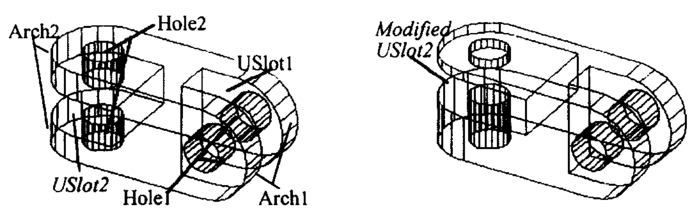
FEATURE-BASED EDITING PROBLEMS AND SOLUTIONS

In this section, further definitions are given to help describe the feature and the part both before and after editing. Following these definitions, potential 3D feature-based editing problems are described and solutions are proposed.

Key definitions and part descriptions

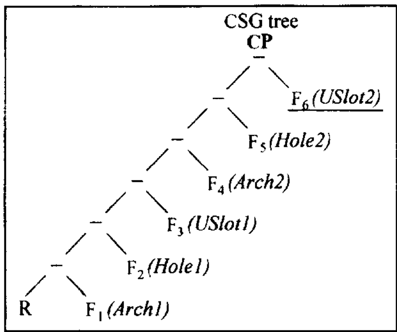
- **S_F(P)**: represents the Set of Features of a part **P**, e.g. $S_F(P)=\{F_1,F_2...,F_n\}$;
- **U(S)**: represents the Union of all features in the feature set **S**. For example, **U(S_F(P))** means the union of all features in $S_F(P)$, i.e. $U(S_F(P)=\{F_1 \cup F_2 \cup ... \cup F_n\}$;
- **CP**: represents the feature expression of a Current Part to be modified;
- **CF**: represents the Current Feature picked from a current part to be modified, $CF \in S_F(CP)$);
- **MP**: represents the feature expression of a Modified Part, which is derived from a current part after editing;
- **MF**: represents the Modified Feature in a modified part. The parameters of a modified feature are derived from those of its corresponding current feature;
- **S_C(MF)**: represents the Set of features excluding the current feature in the current part that are Contained in the **MF**, i.e. $S_C(MF)=\{F_j|F_j \in S_F(CP), F_j \neq CF, F_j \subset MF, j=1,...,n\}$. The contain-relationship, " $\subset$ ", implies the volume enclosure relationship in a 3D space;
- **S_I(F_i)**: represents the Set of features excluding F_i in the current part that are Intersected with **F_i**, e.g. $S_I(CF)=\{Fj|Fj \in S_F(CP),\ F_j \neq CF,\ F_j \cap CF \neq \varnothing,\ j=1,...,n\}$;
- $\setminus$: represents the difference operator defined in the set theory; for example, $\{A,B,C\}\setminus\{B,C\}=\{A\}$.

Using the above-defined terms and implied concepts, we can clearly describe the feature-based editing process below. For the current part with features of arch, hole, and Uslot as shown in *Figure 3a*, when the feature USlot2 is enlarged and moved upward, the corresponding modified part can be derived as shown in *Figure 3b*. The intersected portion of USlot2, Arch2 and Hole2 is changed accordingly. The feature-based



(a) the sample current part (b) the modified part, modified from (a)

Figure 3 Illustrations of the current part (*CP*) and the modified part (*MP*)



Feature list: $F_1(Arch1)$, $F_2(Hole1)$, $F_3(USlot1)$, $F_4(Arch2)$, $F_5(Hole2)$, $F_6(USlot2)$;

$S_F(CP)=\{F_1, F_2, F_3, F_4, F_5, F_6\}$;

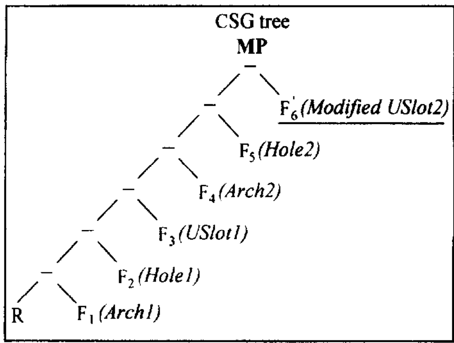
R: block type raw material;

Feature expression: $CP=(((((R-F_1)-F_2)-F_3)-F_4)-F_5)-F_6$;

$CF=F_6(USlot2)$;

$S_I(CF)=\{F_4, F_5\}$.

(a) The information of the current part corresponding to *Figure 3a*



Feature list: $F_1(Arch1)$, $F_2(Hole1)$, $F_3(USlot1)$, $F_4(Arch2)$, $F_5(Hole2)$, $F_6'(Modified\ USlot2)$;

$S_F(MP)=\{F_1, F_2, F_3, F_4, F_5, F_6'\}$;

Feature expression: $MP=(((((R-F_1)-F_2)-F_3)-F_4)-F_5)-F_6'$;

$MF=F_6'(Modified\ USlot2)$ ⇒ locates higher than CF, and its width is larger than that of CF;

$S_C(MF)=\varnothing$.

(b) The information of the modified part corresponding to *Figure 3b*

Figure 4 The feature-based descriptive information corresponding to the current part and the modified part shown in *Figure 3*

descriptive information of the current part and the modified part is given in *Figure 4*.

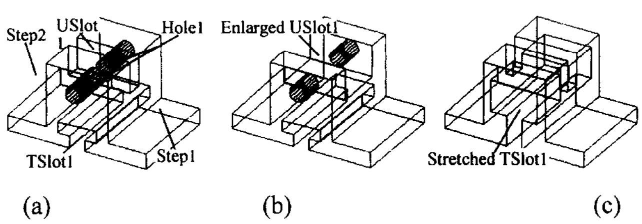
Problem description and solving principle

When a part is edited, two problems for the part description will occur:

(1) The B-rep of the modified part will be changed.
(2) The CSG representation (or feature list) of the modified part will be changed.

For example, in a feature-based design system, the designer wishes to stretch the USlot1 feature in *Figure 5a* to be as it is shown in *Figure 5b*. How can the system derive the modified B-rep efficiently? Or, if the designer wishes to stretch the TSlot1 feature in *Figure 5a* upward such that Hole1 is completely enclosed as shown in *Figure 5c*, how shall the vanished Hole1 be dealt with?

The solving principle is to analyze all possible editing cases first, and then use the properties described in the last section to obtain the correct results for each case. Finally, according to the results, we propose a two-stage feature interaction solving approach.



(a) (b) (c)

Figure 5 Illustrative examples of a 3D feature-based editing problem. (a) A feature-based prismatic part. (b) The result of enlarging the USlot1 which is intersected with Hole1. (c) The result of stretching the TSlot1 which encloses Hole1

Possible editing cases

In each feature-based editing case, the change of the current feature into the modified feature dominates the change of the part. For example, *Figure 5b* illustrates the current feature that may intersect with some other features in the original designed part. While *Figure 5c* illustrates the modified feature that may enclose some other features. The potential editing cases are listed in *Table 1*. The cases are numbered in two portions: the first number is for Stage 1 and the second for Stage 2. In Stage 1, we consider:

(1) The enclosure relationships between the current feature and modified feature.
(2) The intersection relationships between the current feature and the other existing features to update the B-rep of the modified part efficiently.

While in Stage 2, we consider whether any existing features are enclosed by the modified feature, and then process the feature list of the modified part to have it stored more compactly. The updated B-rep can be used to derive the feature list when the five feature interaction properties described previously are considered. *Table 2* demonstrates the examples of potential editing cases and corresponding solving stages.

Derivation of solutions for the feature-based editing problems

To simplify the derivation process, the current feature will be isolated from the feature list of the current part, $S_F(CP)$. Thus, the current part can be represented as:

$$CP = R - F_1 - \ldots - F_n = R - U(S_F(CP))$$
$$= (R - U(S_F(CP)\backslash\{CF\})) - CF, \text{ (by Property 2)} \quad (1)$$

Table 1 Potential feature-based editing cases

Stage 1			Stage 2	
Case no.	Criterion 1	Criterion 2	Case No.	Criterion 3
1	CF is enclosed in MF, i.e. CF ⊂ MF		1.1	No other feature is enclosed in MF, i.e. $S_C(MF) = \varnothing$
			1.2	Some features are enclosed in MF, i.e. $S_C(MF) \neq \varnothing$
2	CF encloses MF, i.e. CF ⊃ MF	No other feature in CP intersects CF, i.e. $S_I(CF) = \varnothing$	2.1	No other feature is enclosed in MF, i.e. $S_C(MF) = \varnothing$
			2.2	Some features are enclosed in MF, i.e. $S_C(MF) \neq \varnothing$
3	CF encloses MF, i.e. CF ⊃ MF	Some features in CP are intersected with CF, i.e. $S_I(CF) \neq \varnothing$	3.1	No other feature is enclosed in MF, i.e. $S_C(MF) = \varnothing$
			3.2	Some features are enclosed in MF, i.e. $S_C(MF) \neq \varnothing$
4	CF and MF are not enclosed in each other, i.e. CF ⊄ MF and MF ⊄ CF	No other feature in CP is intersected with CF, i.e. $S_I(CF) = \varnothing$	4.1	No other feature is enclosed in MF, i.e. $S_C(MF) = \varnothing$
			4.2	Some features are enclosed in MF, i.e. $S_C(MF) \neq \varnothing$
5	CF and MF are not enclosed in each other, i.e. CF ⊄ MF and MF ⊄ CF	No other feature in CP intersects CF, i.e. $S_I(CF) \neq \varnothing$	5.1	No other feature is enclosed in MF, i.e. $S_C(MF) = \varnothing$
			5.2	Some features are enclosed in MF, i.e. $S_C(MF) \neq \varnothing$

The detailed processes for deriving the solutions of the cases 1, 1.1, and 1.2 listed in *Table 1* are described below. Similar derivations for the other cases are given in Appendix A.**Case 1**: $CF \subset MF$(*Stage 1*: B-rep updating.) As the current feature (*CF*) is enclosed in the modified feature (*MF*), only the modified feature can be seen in the modified part. So, the B-rep of the modified part can be obtained by simply subtracting the modified feature from the current part. That is

$$MP = CP - MF \tag{2}$$

(*Stage 2*: Feature list simplification.)

Case 1.1: $CF \subset MF$ and $S_C(MF) = \varnothing$

$$MP = CP - MF,$$

following Case 1 and by Equation 2

$$= [(R - U(S_F(CP)\backslash\{CF\})) - CF] - MF$$

rewriting the CP term by Equation 1

$$= (R - U(S_F(CP)\backslash\{CF\})) - MF,$$

by Property 1, removing the enclosed feature *CF*. So, $S_F(MP) = \{(S_F(CP)\backslash\{CF\}), MF\}$. That is, the feature list of the modified part, $S_F(MP)$, can be obtained from the feature list of the current part, $S_F(CP)$, by replacing the current feature *CF* with the modified feature *MF*.

Case 1.2 $CF \subset MF$ and $S_C(MF) \neq \varnothing$

$$MP = (R - U(S_F(CP)\{\{CF\})) - MF$$

following Case 1.1.

$$= (R - U(S_F(CP)\backslash\{CF\}\backslash S_C(MF))) - MF$$

by Property 1, removing the features enclosed in the *MF*.

So, $S_F(MP) = \{(S_F(CP)\backslash\{CF\}\backslash S_C(MF)), MF\}$. That is, the $S_F(MP)$ can be obtained from the feature list of the current part, $S_F(CP)$, by:

(1) Removing the features which are enclosed in the modified feature.
(2) Adding the modified feature.

For example, to the case 1.1. of *Table 2*, we enlarge the width of USlot1 (the current feature) such that it is enclosed by the new feature USlot1′ (the modified feature). Then, the modified B-rep can be evaluated directly by subtracting the modified feature from the original B-rep of the current part. Because no other existing features are enclosed by the USlot1′, the final feature list can be obtained by replacing the original USlot with USlot1′ in the original feature list of the part. As for the case 1.2, the B-rep of the modified part can be evaluated similarly. However, when we enlarge the width and height of USlot1 (the current feature), the features of Hole1, Hole2, and USlot1 will be enclosed in the newly derived feature USlot1′ (the modified feature). Thus, the final feature list can be obtained by replacing the feature USlot1 with USlot1′, and by removing the enclosed features of Hole1 and Hole2.

Resulting rules for the two-stage feature interaction solving approach

The results derived for the feature-based editing cases are summarized in *Table 3*. The illustrative examples corresponding to the feature-based editing cases in *Table 2* are given in *Table 4*.

Analysing the final derivation results in *Table 3*, we find that in Stage 1 the operations to update the B-rep of a designed part can be simplified into three groups:

(1) "$CP - MF$" (case 1).
(2) "$(CP + CF) - MF$" (cases 2 and 4).
(3) "$((CP + CF) - U(S_I(CF))) - MF$" (cases 3 and 5).

Also, in Stage 2, the feature list can be simplified into two groups: "$\{(S_F(CP)\backslash\{CF\}), MF\}$" (cases 1.1, 2.1, 3.1, 4.1, and 5.1) and "$\{(S_F(CP)\backslash\{CF\}\backslash S_C(MF)), MF\}$" (cases 1.2, 4.2, and 5.2). So, the general rules for solving the feature-based editing problems can be simplified as follows:

Stage 1: updating the B-rep.

The modified part can be derived from the current part by determining whether:

(1) The current feature is enclosed in the modified feature.
(2) The $S_I(CF)$ is empty.

So, in Stage 1, the rules can be summarized:

If $CF \subset MF$

Table 2 Detailed description of the potential of the potential feature-based editing problems

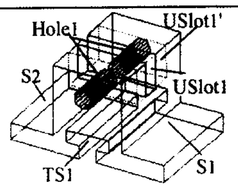
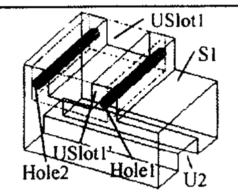
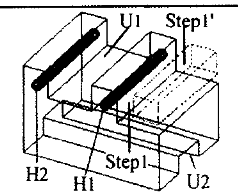
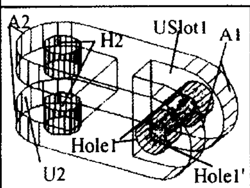
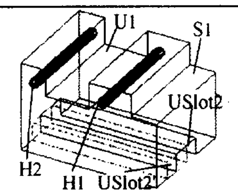
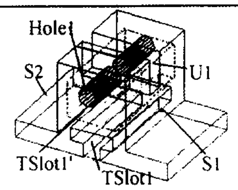
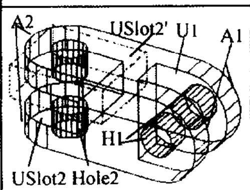
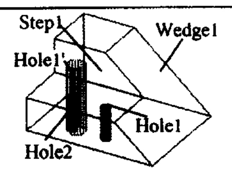
Stage 1		Stage 2		Illustrative examples						
case no.	criteria 1 and 2	case no.	criterion 3	designed part	$S_F(CP)$	CF	MF	modified parameters	$S_I(CF)$	$S_C(MF)$
1	$CF \subset MF$	1.1	$S_C(MF)=\varnothing$		Hole1, Step1, Step2, TSlot1, USlot1	USlot1	USlot1'	enlarge the width	Hole1	NULL
		1.2	$S_C(MF)\neq\varnothing$		Hole1, Hole2, Step1, USlot1, USlot2	USlot1	USlot1'	extend the width and height	NULL	Hole1, Hole2
2	$CF \supset MF$ and $S_I(CF)=\varnothing$	2.1	$S_C(MF)=\varnothing$		Hole1, Hole2, Step1, USlot1, USlot2	Step1	Step1'	shrink the width and height	NULL	NULL
		2.2	$S_C(MF)\neq\varnothing$	will not occur						
3	$CF \supset MF$ and $S_I(CF)\neq\varnothing$	3.1	$S_C(MF)=\varnothing$		Arch1, Arch2, Hole1, Hole2, USlot1, USlot2	Hole1	Hole1'	shrink the diameter	USlot1	NULL
		3.2	$S_C(MF)\neq\varnothing$	will not occur						
4	$CF \not\subset MF$ and $MF \not\subset CF$ and $S_I(CF)=\varnothing$	4.1	$S_C(MF)=\varnothing$		Hole1, Hole2, Step1, USlot1, USlot2	USlot2	USlot2'	move the operation point to the left	NULL	NULL
		4.2	$S_C(MF)\neq\varnothing$		Hole1, Step1, Step2, TSlot1, USlot1	TSlot1	TSlot1'	stretch the width and height	NULL	Hole1
5	$CF \not\subset MF$ and $MF \not\subset CF$ and $S_I(CF)\neq\varnothing$	5.1	$S_C(MF)=\varnothing$		Arch1, Arch2, Hole1, Hole2, USlot1, USlot2	USlot2	USlot2'	move up the operation point and enlarge the width	Hole2	NULL
		5.2	$S_C(MF)\neq\varnothing$		Hole1, Hole2, Step1, Wedge1	Hole1	Hole1'	move the operation point to the left and enlarge the diameter	Wedge1, Step1	Hole2

Table 3 The derivation results for the feature-based editing cases

Stage 1		MP for B-rep updating	Stage 2		MP for feature list simplifications: $S_F(MP)$
Case no.	Criteria 1 and 2		Case no.	Criterion 3	
1	$CF \subset MF$	$CP - MF$	1.1	$S_C(MF) = \varnothing$	$\{(S_F(CP)\backslash\{CF\}),MF\}$
			1.2	$S_C(MF) \neq \varnothing$	$\{(S_F(CP)\backslash\{CF\}\backslash S_C(MF)),MF\}$
2	$CF \supset MF$ and $S_I(CF) = \varnothing$	$CP + CF - MF$	2.1	$S_C(MF) = \varnothing$	$\{(S_F(CP)\backslash\{CF\}.MF\}$
			2.2	$S_C(MF) \neq \varnothing$	(infeasible)
3	$CF \supset MF$ and $S_I(CF) \neq \varnothing$	$CP + CF - U(S_I(CF)) - MF$	3.1	$S_C(MF) = \varnothing$	$\{(S_F(CP)\backslash\{CF\}),MF\}$
			3.2	$S_C(MF) \neq \varnothing$	(infeasible)
4	$CF \not\subset MF$ and $MF \not\subset CF$ and $S_I(CF) = \varnothing$	$CP + CF - MF$	4.1	$S_C(MF) = \varnothing$	$\{(S_F(CP)\backslash\{CF\}),MF\}$
			4.2	$S_C(MF) \neq \varnothing$	$\{(S_F(CP)\backslash\{CF\}\backslash S_C(MF)),MF\}$
5	$CF \not\subset MF$ and $MF \not\subset CF$ and $S_I(CF) \neq \varnothing$	$CP + CF - U(S_I(CF)) - MF$	5.1	$S_C(MF) = \varnothing$	$\{(S_F(CP)\backslash\{CF\}),MF\}$
			5.2	$S_C(MF) \neq \varnothing$	$\{(S_F(CP)\backslash\{CF\}\backslash S_C(MF)),MF\}$

then $MP = CP \underline{- MF}$

else

If $S_I(CF) = \varnothing$

then $MP = (CP \underline{+ CF) - MF}$

else $MP = ((CP \underline{+ CF) - U(S_I(CF))) - MF}$

The underlined terms are the operations for updating the B-rep. It is seen that the procedures and the time to evaluate the Boolean operations are greatly reduced. Only when some features in the current part are intersected with the current feature (i.e. $S_I(CF) \neq \varnothing$) should we consider the features other than the current feature and modified feature in the re-evaluation process.

Stage 2: simplifying the feature list.

The concise feature list of a modified part, $S_F(MP)$, can be obtained by checking whether any feature in the current part excluding the current feature is enclosed in the modified feature. Thus, in Stage 2, the rules can be simplified as:

If $S_C(MF) = \varnothing$

then $S_F(MP) = \{(S_F(CP)\backslash\{CF\}), MF\}$

else $S_F(MP) = \{(S_F(CP)\backslash\{CF\}\underline{\backslash S_C(MF)}), MF\}$

The set of current feature, $\{CF\}$, in the feature list is replaced by the modified feature; the redundant features enclosed in the modified feature, $S_C(MF)$, are eliminated. The solutions for each case of the feature-based editing problems are given in the next sub-section.

Feature-based editing functions of the two-stage approach

For handling individual feature-based editing functions of moving, shrinking, stretching, deleting, and adding, the rules of the two-stage feature interaction solving approach are described as follows:

(1)If EditCommand = Move, or
EditComand = Shrink, or
EditCommand = Stretch

Stage 1:updating the B-rep after the move, shrink, or stretch operation.

If $CF \subset MF$

then $MP = CP \underline{- MF}$

else

if $S_I(CF) = \varnothing$

then $MP = (CP \underline{+ CF) - MF}$

else $MP + ((CP \underline{+ CF) - U(S_I(CF))) - MF}$

Stage 2: simplifying the feature list after the move, shrink, or stretch operation.

If $S_C(MF) = \varnothing$

then $S_F(MP) = \{S_F(CP)\backslash\{CF\}), MF\}$

else $S_F(MP) = \{(S_F(CP)\backslash\{CF\}\underline{\backslash S_C(MF)}), MF\}$

(2)If EditCommand = Delete

Stage 1: updating the B-rep after the delete operation.

If $S_I(CF) = \varnothing$

then $MP = CP \underline{+ CF}$

else $MP = (CP \underline{+ CF) - U(S_I(CF))}$

Stage 2: simplifying the feature list after the delete operation.

$S_F(MP) = \{(S_F(CP)\backslash\{CF\})\}$

(3)If EditCommand = Add

Stage 1: updating the B-rep after the add operation.

$MP = CP \underline{- CF}$

Stage 2: simplifying the feature list after the add operation.

If $S_C(MF) = \varnothing$

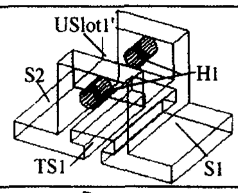
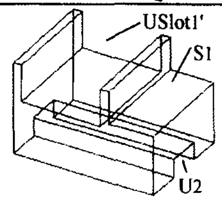
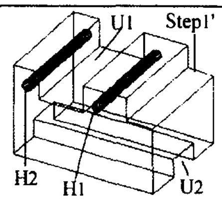
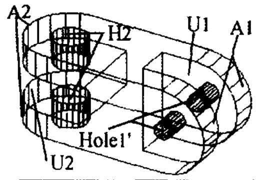
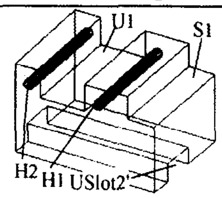
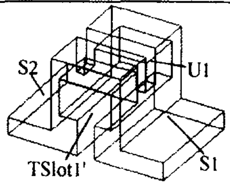
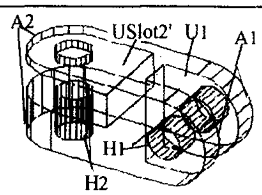
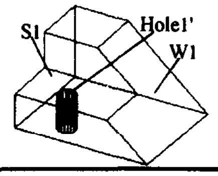
then $S_F(MP) = \{(S_F(CP), \{CF\})\}$

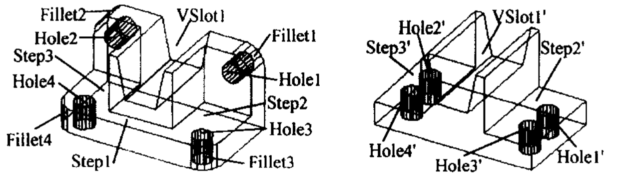
else $S_F(MP) = \{(S_F(CP)\underline{\backslash S_C(CF)}), MF\}$

COMPUTER SIMULATION

The computer interaction solving approach for 3D feature-based part editing problems is implemented on a PC/386

Table 4 Illustrations for the feature-based editing cases in Table 2, including the final display graph and feature list

Stage 1		Stage 2		Derived results corresponding to *Table 2*	
case no.	criteria 1 and 2	case no.	criterion 3	designed part	SF(MP)
1	CF⊂MF	1.1	Sc(MF)=∅		Hole1, Step1, Step2, TSlot1, USlot1'
		1.2	Sc(MF)≠∅		Step1, USlot1', USlot2
2	CF⊃MF and SI(CF)=∅	2.1	Sc(MF)=∅		Hole1, Hole2, Step1', USlot1, USlot2
3	CF⊃MF and SI(CF)≠∅	3.1	Sc(MF)=∅		Arch1, Arch2, Hole1', Hole2, USlot1, USlot2
4	CF⊄MF and MF⊄CF and SI(CF)=∅	4.1	Sc(MF)=∅		Hole1, Hole2, Step1, USlot1, USlot2'
		4.2	Sc(MF)≠∅		Step1, Step2, TSlot1', USlot1
5	CF⊄MF and MF⊄CF and SI(CF)≠∅	5.1	Sc(MF)=∅		Arch1, Arch2, Hole1, Hole2, USlot1, USlot2'
		5.2	Sc(MF)≠∅		Hole1', USlot1, Wedge1



(a) the original part (b) the objective part

Figure 6 The illustrative parts for the proposed feature-based editing approach

with the PC/MS-Windows[7]. The ACIS solid modelling system[15] and language MetaWare High C/C++ are used for the development of a user-friendly interface. The feature icons can be selected and positioned dynamically. When 3D-prismatic-parts are designed through the feature-based part design system[9], this editing approach is implemented as a further modification module.

We use one example to show the functions of the proposed editing approach in detail. *Figure 6b* is the objective example to be modified from the sample part shown in *Figure 6a*.

Using the feature-based part design system, we can create the original sample part as shown in *Figure 7*.

The procedures and corresponding results for modifying the sample part are shown in *Table 5*. The feature list of the current sample part, $S_F(CP)$, contains: Fillet1, Fillet2, Fillet3, Fillet4, Hole1, Hole2, Hole3, Hole4, Step1, Step2, Step3, and VSlot1. In *Table 5*, the editing functions including moving deleting, stretching, and adding are used in deriving the part. The features of Hole1 and Hole2 are automatically removed by the stretching operations of processes 7 and 8.

Two more examples which are obtained by similar design and editing procedures are shown in *Figure 8a* and *b*. There are nineteen features in *Figure 8a* and fourteen features in *Figure 8b*. It would normally take only about twenty minutes to complete the design process for each.

CONCLUSIONS AND SUGGESTIONS

In this paper, a two-stage feature interaction solving approach for 3D feature-based part editing problems is proposed and developed. The possible enclosure and

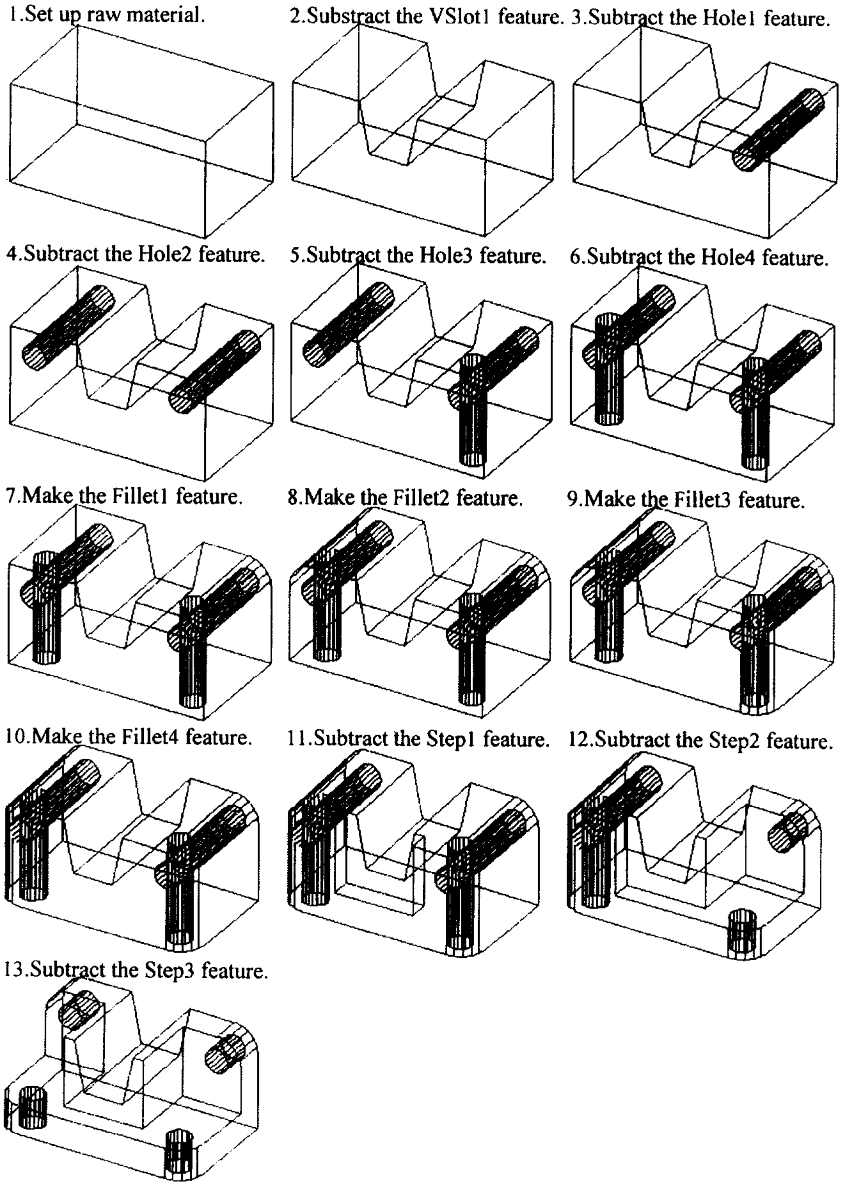
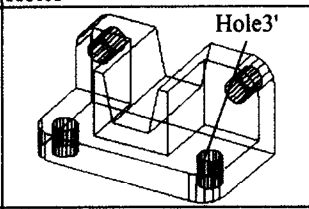
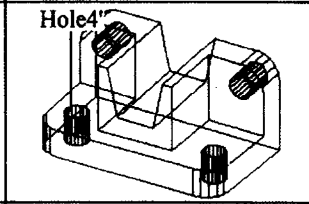
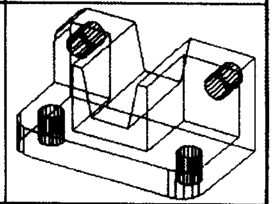
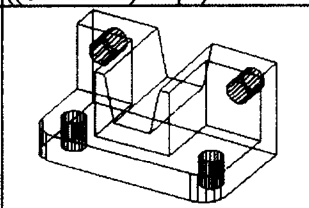
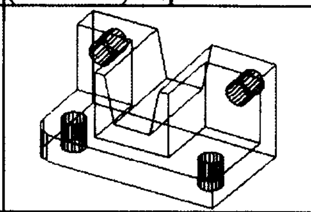
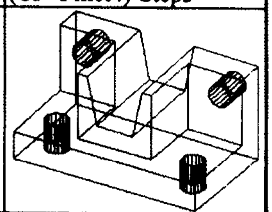
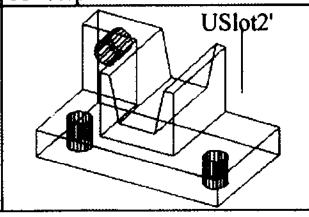
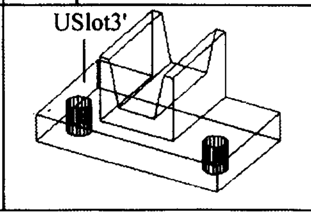
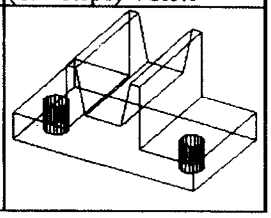


Figure 7 The original sample part created by the system of Perng and Chang[15] in thirteen steps

Table 5 The procedures and corresponding results for modifying the sample part in *Figure* 6a into the objective part in *Figure* 6b

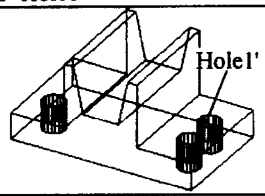
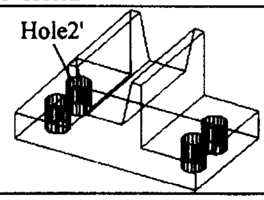
Editing process	1. Move Hole3	2. Move Hole4	3. Delete Fillet1
CF	Hole3	Hole4	Fillet1
MF	Hole3'	Hole4'	
Modified parameters	Move the operation point of Hole3 from the front end to the center	Move the operation point of Hole4 from the front end to the center	
$S_I(CF)$	{Step2, Hole1}	{Step3, Hole2}	{Step2, Fillet3}
$S_C(MF)$	∅	∅	∅
Boolean operations	(((CP+Hole3)-Step2)-Hole1)-Hole3'	(((CP+Hole4)-Step3)-Hole1)-Hole4'	((CP+Fillet1)-Step2)-Fillet3
B-rep result			
$S_F(MP)$	Fillet1, Fillet2, Fillet3, Fillet4, Hole1, Hole2, **Hole3'**, Hole4, Step1, Step2, Step3, and VSlot1	Fillet1, Fillet2, Fillet3, Fillet4, Hole1, Hole2, Hole3', **Hole4'**, Step1, Step2, Step3, and VSlot1	Fillet2, Fillet3, Fillet4, Hole1, Hole2, Hole3', Hole4', Step1, Step2, Step3, and VSlot1
Editing process	4. Delete Fillet2	5. Delete Fillet3	6. Delete Fillet4
CF	Fillet2	Fillet3	Fillet4
MF			
Modified parameters			
$S_I(CF)$	{Step3, Fillet4}	{Step2}	{Step3}
$S_C(MF)$	∅	∅	∅
Boolean operations	((CP+Fillet2)-Step3)-Fillet4	(CP+Fillet3)-Step2	(CP+Fillet4)-Step3
B-rep result			
$S_F(MP)$	Fillet3, Fillet4, Hole1, Hole2, Hole3', Hole4', Step1, Step2, Step3, and VSlot1	Fillet4, Hole1, Hole2, Hole3', Hole4', Step1, Step2, Step3, and VSlot1	Hole1, Hole2, Hole3', Hole4', Step1, Step2, Step3, and VSlot1
Editing process	7. Stretch Step2	8. Stretch Step3	9. Delete Step1
CF	Step2	Step3	Step1
MF	Step2'	Step3'	
Modified parameters	Stretch the depth of Step2	Stretch the depth of Step3	
$S_I(CF)$	{Hole1, Hole3'}	{Hole2, Hole4'}	
$S_C(MF)$	{Hole1}	{Hole2}	
Boolean operations	CP-Step2'	CP-Step3'	(CP+Step1)-VSlot1
B-rep result			

intersection cases of 3D features of a designed part are discussed in detail and a set of compact solution rules for these cases are proposed and implemented. The editing functions of moving, shrinking, stretching, deleting, and adding are fully-supported in the proposed editing approach. With the help of this 3D feature-based part editing approach, users can easily and efficiently modify an existing part for further applications.

The characteristics of the proposed feature interaction solving approach are summarized below.

- The B-rep of a modified part can be evaluated with the minimum Boolean operations.
- The feature description of a final designed part can be maintained in a concise form without redundant features.
- The part design/editing system can be operated more quickly and can save more storage spaces.

Table 5 Continued

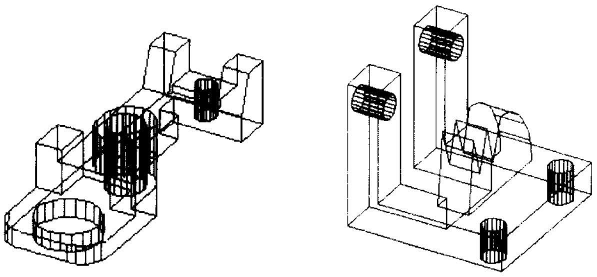
$S_F(MP)$	Hole2, Hole3', Hole4', Step1, **Step2'**, Step3, and VSlot1	Hole3', Hole4', Step1, Step2', **Step3'**, and VSlot1	Hole3', Hole4', Step2', Step3', and VSlot1
Editing process	**10. Add Hole1'**	**11. Add Hole2'**	**End.**
CF	Hole1'	Hole2'	
MF			
Modified parameters			
$S_I(CF)$			
$S_C(MF)$			
Boolean operations	CP-Hole1'	CP-Hole2'	
B-rep result			
$S_F(MP)$	**Hole1'**, Hole3', Hole4', Step1, Step2', Step3, and VSlot1	Hole1', **Hole2'**, Hole3', Hole4', Step1, Step2', Step3', and VSlot1	

However, the part is assumed consist of subtractive volume-features only, a number of issues about 3D feature-based editing still deserve further explorations.

- Detecting and processing more complex cases of united enclosure, such as the case shown in *Figure 9*.
- Extending the part description range by including more types of features, especially protrusive ones.

ACKNOWLEDGEMENTS

This work is supported by the National Science Council, Taiwan, R.O.C., under contracts NSC 83-0415-E-009-004 and NSC 84-2213-E-009-057.



(a) (b)

Figure 8 Two examples that can be obtained by the proposed feature-based design and editing approach

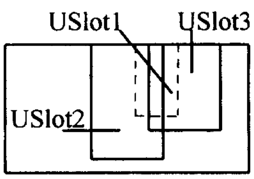


Figure 9 Illustration of united enclosure (front view): the USlot1 is enclosed by the union of the USlot2 and the USlot3. Only one feature of USlot2 or USlot3 does not enclose the USlot1

APPENDIX THE DERIVATION PROCESSES FOR THE FEATURE-BASED EDITING CASES 2–5 OF *TABLE 2*

Case 2: $CF \supset MF$ and $S_I(CF) = \varnothing$

(*Stage 1*: B-rep updating.)

The B-rep of the modified part can be derived by first uniting the current feature with the current part, then subtracting the modified feature from it. Because the current feature is not intersected with any F_i in the current part, there is no need to consider the other features.

So,

$$MP = CP + CF - MF \qquad (3)$$

(*Stage 2*: Feature list simplification.)

Case 2.1.: $CF \supset MF$, $S_I(CF) = \varnothing$ and $S_C(MF) = \varnothing$

$$MP = (CP + CF) - MF$$

following Case 2 and by Equation 3

$$= ([R - U(S_F(CP)\backslash\{CF\})) - CF] + CF) - MF$$

rewriting the CP term by Equation 1

$$= (R - U(S_F(CP)\backslash\{CF\})) - MF$$

by $S_I(CF) = \varnothing$ and Property 4, removing the addition and subtraction terms about the CF

So, $S_F(MP) = \{S_F(CP)\backslash\{CF\}), KF\}$. The result is the same as in Case 1.1.

Case 2.2.: $CF \supset MF$, $S_I(CF) = \varnothing$ and $S_C(MF) \neq \varnothing$

Because the current feature (CF) does not enclose any feature (F_i) in the current part, and the current feature encloses the modified feature, i.e. $CF \supset MF$

$$Sc(MF) = \varnothing$$

This contradicts the given condition: $Sc(MF) \neq \varnothing$. Therefore, this case will not occur.

Case 3.: $CF \supset MF$ and $S_I(CF) \neq \phi$.

(*Stage 1*: B-rep updating.)

The B-rep of the modified part can be obtained by first uniting the current feature, then subtracting the union of $S_I(CF)$ and the modified feature from the current part. Because the current feature only intersects some features in the current part, there is no need to consider the other features which are not in the $S_I(CF)$.

So

$$MP = ((CP + CF) - U(S_I(CF))) - MF \quad (4)$$

(*Stage 2*: Feature list simplification.)

Case 3.1.: $CF \supset MF$, $S_I(CF) \neq \phi$ and $S_C(MF) = \phi$

$$MP = ((CP + CF) - U(S_I(CF))) - MF$$

following Case 3 and by Equation 4

$$= (([(R - U(S_F(CP)\backslash\{CF\}((-CF]$$

$$+ CF) - U(S_I(CF))) - MF$$

rewriting the CP term by Equation 1

$$= ((R - U(S_F(CP)\backslash\{CF\})) - U(S_I(CP))) - MF$$

by $S_I(CF) \neq \varnothing$ and property 5, removing the addition and subtraction terms about the CF

$$= (R - U(S_F(CP)\backslash\{CF\})) - MF$$

by $S_C(MF) = \phi$ and Property 3, removing the features in $U(S_I(CP))$ from the expression.

So, $S_F(MP) = \{(S_F(CP)\backslash\{CF\}), MF\}$.

The result is the same as in Case 1.1.

Case 3.2.: CF^{MF}, $S_I(CF) \neq \varnothing$ and $S_C(MF) \neq \varnothing$

Because the current feature does not enclose any F_i in the current part and CF^{MF},

$$S_C(MF) = \varnothing.$$

This contradicts the given condition: $S_C(MF) \neq \varnothing$. Therefore, this case will not occur.

Case 4.: $CF \not\subset MF$, $MF \not\subset CF$, and $S_I(CF) = \varnothing$

(*Stage 1*: B-rep updating.)

Similar to Case 2, the B-rep of the modified part can be obtained as

$$MP = CP + CF - MF \quad (5)$$

(*Stage 2*: Feature list simplification.)

Case 4.1.: $CF \not\subset MF$, $MF \not\subset CF$, $S_I(CF) = \varnothing$, and $S_C(MF) = \varnothing$

$$MP = CP + CF - MF,$$

following case 4 and by Equation 5

$$= ([(R - U(S_F(CP)\backslash\{CF\})) - CF] + CF) - MF,$$

rewriting the CP term by Equation 1

$$= (R - U(S_F(CP)\backslash\{CF\})) - MF,$$

by $S_I(CF) = \varnothing$ and Property 4, removing the addition and subtraction terms about the CF

So, $S_F(MP) = \{(S_F(CP)\backslash\{CF\}), MF\}$. The result is the same as in Case 1.1.

Case 4.2.: $CF \not\subset MF$, $MF \not\subset CF$, $S_I(CF) = \varnothing$, and $S_C(MF) \neq \varnothing$

$$MP = (R - U(S_F(CP)\backslash\{CF\})) - MF$$

following Case 4.1

$$= (R - U(S_F(CP)\backslash\{CF\}\backslash S_C(MF))) - MF$$

by Property 1, removing the features enclosed in the MF

So, $S_F(MP) = \{(S_F(CP)\backslash\{CF\}\backslash S_C(MF)), MF\}$. That is, the redundant features in the $S_C(MF)$ are removed. The result is the same as Case 1.2.

Case 5.: $CF \not\subset MF$, $MF \not\subset CF$, and $S_I(CF) \neq \varnothing$

(*Stage 1:* B-rep updating.)

Similar to Case 3, the B-rep of the modified part can be obtained as

$$MP = ((CP + CF)U(S_I(CF))) - MF \quad (6)$$

(*Stage 2:* Feature list simplification.)

Case 5.1.: $CF \not\subset MF$, $MF \not\subset CF$, and $S_I(CF) \neq \varnothing$, and $S_C(MF) = \varnothing$

$$MP = ((CP + CF) - U(S_I(CF))) - MF$$

following Case 5 and by Equation 6

$$= (([R - U(S_F(CP)\backslash\{CF\})) - CF] + CF) - U(S_I(CF))) - MF,$$

rewriting the CP term by Equation 1

$$= ((R - U(S_F(CP)\backslash\{CF\})) - U(S_I(CF))) - MF,$$

by $S_I(CF) \neq \varnothing$ and Property 5, removing the addition and subtraction terms about the CF

$$= (R - U(S_F(CP)\backslash\{CF\})) - MF,$$

by $S_C(MF) = \varnothing$ and Property 3, removing the features in $U(S_I(CP))$ from the expression.

So, $S_F(MP) = \{(S_F(CP)\backslash\{CF\}), MF\}$. The result is the same as in Case 1.1.

Case 5.2.: $CF \not\subset MF$, $MF \not\subset CF$, and $S_I(CF) \neq \varnothing$, and $S_C(MF) \neq \varnothing$

$$MP = (R - U(S_F(CP)\backslash\{CF\})) - MF,$$

following Case 5.1

$$= (R - U(S_F(CP)\backslash\{CF\}\backslash S_C(MF))) - MF,$$

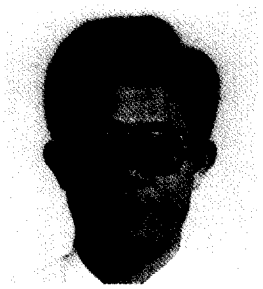
by Property 1, removing the features enclosed in the MF

So, $S_F(MP) = \{(S_F(CP)\backslash\{CF\}\backslash S_C(MF)), MF\}$.The result is the same as in Case 1.2.

REFERENCES

1. Case, K. and Gao, J., Feature technology: an overview. *International Journal of Computer Integrated Manufacturing*, 1993, **6**(1 and 2), 2–12.
2. Chan, S. C. and Voelcker, H. B., An introduction to MPL — a new machining process/programming language. In *Proceedings of the 1986 IEEE International Conference on Robotics and Automation*, San Francisco, CA, **1**, 7–10 April 1986, pp. 333–344.
3. Dixon, R. J., Libardi, E. F. Jr. and Nielsen, D. H., Unresolved research issues in development of design-with-features systems. In *Geometric Modelling for Product Engineering*, ed. M. J. Wozny, J. U. Tuner and K. Preiss. Elsevier Science Publishers B V, North Holland, IFIP, 1990, pp. 183–196.
4. Jasthi, S. R. K., Prasad, A. V. S. R. K., Manidhar, G., Rao, P. N., Rao, U. R. K. and Tewari, N. K., A feature-based part description system for computer-aided process planning. *Journal of Design and Manufacturing*, 1994, **4**(1), 67–80.
5. Lim, S. S., Lee, I. B. H., Lim, L. E. N. and Ngoi, B. K. A., Computer-aided concurrent design of product and assembly processes: a literature review. *Journal of Design and Manufacturing*, 1995, **5**(2), 67–88.
6. Mayer, H., Su, C. J., Sun, T.-L. and Wysk, R. A., ECTOF: a feature representation technique for concurrent engineering applications. *Journal of Design and Manufacturing*, 1995, **4**(1), 49–65.
7. Microsoft Corporation, *Microsoft Windows 3.1 Guide to Programming*. Washington Microsoft Press, Redmond, 1992.
8. Perng, D. B., Chen, Z. and Li, R. K., Automatic 3D machining feature extraction from 3D CSG solid input. *Computer-Aided Design*, 1990, **22**(5), 285–295.

9. Perng, D. B. and Chang, C. F., A new feature-based design system with dynamic-editing. *Computers and Industrial Engineering.* Accepted for publication.
10. Pratt, M. J., Synthesis of an optimal approach to form feature modelling. In *Proceedings of the 1988 ASME International Computers in Engineering Conference and Exhibition*, San Francisco, CA, **1**, July/August 1988, pp. 263–274.
11. Rossignac, J. R. and Voelcker, H. B., Active zones in CSG for accelerating boundary evaluation, redundancy elimination, inference detection, and shading algorithms. *ACM Transactions on Graphics*, 1989, **8**(1), 51–87.
12. Rossignac, J. R., Issues on feature-based editing and interrogation of solid models. *Computer and Graphics*, 1990, **14**(2), 149–172.
13. Rossignac, J. R. and O'Connor, M. A., SGC: A dimension-independent model for point sets with internal structures and incomplete boundaries. In *Geometric Modelling for Product Engineering*, ed. M. J. Wozny, J. U. Tuner and K. Preiss. Elsevier Science Publishers B.V., North Holland, IFIP, 1990, pp. 145–180.
14. Salomons, O. W., van Houten, F. J. A. M. and Kals, H. J. J., Review of research in feature-based design. *Journal of Manufacturing Systems*, 1993, **12**(2), 113–132.
15. Spatial Technology Inc, *ACIS Geometric Modeller Interface Guide*, 2425 55th Street, Building A, Boulder, CO 80301, USA, 1992.
16. Suh, H. and Ahluwalia, R. S., Feature modification in incremental feature generation. *Computer-Aided Design*, 1995, **27**(8), 627–635.
17. Su, C. J., Mayer, R. J., Sun, T.-L. and Wysk, R. A. A., Three-phase method for feature interaction resolution. *Journal of Design and Manufacturing*, 1994, **4**(2), 153–166.
18. Tilove, R. B., Set Memberships Classification: A unified approach to geometric intersection problems. *IEEE Transactions on Computers* , 1980, **C-29**(10), 874–883.
19. Turner, G. P. and Anderson, D. C., An object-oriented approach to interactive, feature-based design for Quick Turnaround Manufacturing. In *Proceedings of the 1988 ASME International Computers in Engineering Conference and Exhibition*, San Francisco, CA, **1**, July/August 1988, pp. 551–555.



Dr Der-Baau Perng was born in 1955 in Taiwan, R.O.C. He received the BS and MS and PhD degrees all in Computer Engineering from National Chiao Tung University, Hsin chu, Taiwan, R.O.C., in 1979, 1981 and 1988, respectively. He was on the faculty of the Department of Computer Engineering (1983–1984) at National Chiao Tung University. He joined the Department of Industrial Engineering and Management at National Chiao Tung University from 1988 and was the Chairman from 1991 to 1994. Dr Perng is currently a Professor of the Department of Industrial Engineering and Management and the Director of the Production System Automation Research Center, National Chiao Tung University. His current research interests include computer-aided design, computer-aided industrial design, solid modelling systems, reverse engineering and computer vision.

Chao-Fan Chang is currently working towards a PhD in computer-aided design systems at the Institute of Industrial Engineering, National Chiao Tung University, Hsin chu, Taiwan, R.O.C. He obtained a BS in 1992 from the Department of Industrial Engineering and Management, National Chiao Tung University. His research interests are in the areas of feature-based design/editing, automatic assembly modelling and solid modelling.