Feature-Based Multi-Resolution Modeling of Solids Using History-Based Boolean Operations — Part I : Theory of History-Based Boolean Operations —

Sang Hun Lee*
School of Mechanical and Automotive Engineering, Kookmin University, Seoul 136-702, Korea
Kyu-Yeul Lee
Department of Naval Architecture and Ocean Engineering, Seoul National University, Seoul 151-744, Korea
Yoonwhan Woo
Graduate School of Automotive Engineering, Kookmin University, Seoul 136-702, Korea
Kang-Soo Lee
School of Mechanical Engineering, Hanbat National University, Daejeon 305-719, Korea

The requirements of multi-resolution models of feature-based solids, which represent an object at many levels of feature detail, are increasing for engineering purposes, such as analysis, network-based collaborative design, virtual prototyping and manufacturing To provide multi-resolution models for various applications, it is essential to generate adequate solid models at varying levels of detail (LOD) after feature rearrangement, based on the LOD criteria However, the non-commutative property of the union and subtraction Boolean operations is a severe obstacle to arbitrary feature rearrangement To solve this problem we propose history-based Boolean operations that satisfy the commutative law between union and subtraction operations by considering the history of the Boolean operations Because these operations guarantee the same resulting shape as the original and reasonable shapes at the intermediate LODs for an arbitrary rearrangement of its features, various LOD criteria can be applied for multi-resolution modeling in different applications

Key Words : Non-Manifold, Solid, Multi-Resolution, Feature, Boolean Operation

1. Introduction

In the area of computer graphics, extensive research on multi-resolution modeling and its applications has been carried out to enable fast display (Schröder et al, 1992, Cignoni et al, 1998) The objects for multi-resolution modeling have in the main been polyhedral models, including triangles, and various polygonal simplification methods such as edge-collapse and vertex-removal have been developed to provide models at the required level of detail (LOD) The applications are mainly fast rendering and transmission of geometric models in computer graphics The objects for removal, or suppression, to generate low-resolution models are the lower levels of the topological entities, such as vertices, edges, or faces

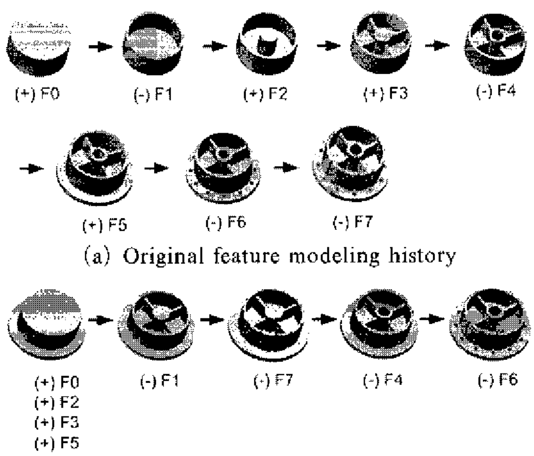
Unlike the conventional polyhedral approaches, multi-resolution modeling of the feature-based B-rep solid models has only recently been

* Corresponding Author,
E-mail shlee@kookmin ac.kr
TEL +82-2-910-4835, **FAX** +82-2-916-0701
School of Mechanical and Automotive Engineering, Kookmin University, Seoul 136-702, Korea (Manuscript **Received** June 28, 2004, **Revised** November 11, 2004)

studied (Choi et al., 2002 ; Koo and Lee, 2002 ; Kim et al., 2003 ; Lee S. H. et al., 2002 ; Lee J. Y. et al., 2002 ; Lee et al., 2004). Here, the object of multi-resolution modeling is a solid model and the suppressed objects are form features that are at an even higher level of modeling entities than the topological entities. Features are classified into two groups : additive and subtractive features. In the previous research, feature-based multi-resolution modeling algorithms have been developed based on the assumption that, as illustrated in Fig. 1, the model at the lowest resolution is constructed by uniting all the additive features and the models at higher resolutions are generated by applying subtractive features in descending order of volume. Therefore, if the features are rearranged in arbitrary order, previous research methods do not necessarily result in the same shape as the original solid model.

However, to apply multi-resolution modeling techniques to various applications, it is essential to include additive features for intermediate LOD models. A severe obstacle for this task is the non-commutative property of the union and subtraction Boolean operations. To solve this problem we propose history-based Boolean operations, based on the merge-and-select algorithm (Crocker and Reinke, 1991 ; Masuda 1992 ; Kim et al., 1996). Unlike the conventional Boolean operations, they satisfy the commutative law for union and subtraction operations by considering the history of Boolean operations. Therefore, these operations guarantee the same resultant shape as the original and reasonable shapes at intermediate LODs for an arbitrary rearrangement of features. As a result, various LOD criteria can be applied for multi-resolution modeling in different applications.

The remainder of the paper is organized as follows. Section 2 defines the problem. Section 3 introduces the definition of history-based Boolean operations. Section 4 discusses the commutative property of history-based Boolean operations. Section 5 introduces the adaptation of the history-based Boolean operations for more acceptable intermediate LOD models. Section 6 presents our conclusions.



(a) Original feature modeling history

(b) Rearranged subtractive features according to their volume

Fig. 1 An example of feature-based multi-resolution modeling proposed by Choi et al. (2002)

2. Problem Definition

Feature-based modelers use a modeling method in which a feature is a basic modeling unit, and an object is modeled by adding features incrementally to a basic shape feature (Shah, 1995 ; Lee, 1999). According to Part 48 of STEP, form features are classified into three basic types : volume, transition, and pattern features (Dunn, 1992). A volume feature is an increment, or decrement, to the volume of a shape, such as a hole or a boss. A transition feature separates or blends surfaces, such as fillets or chamfers. A feature pattern is a set of similar features in a

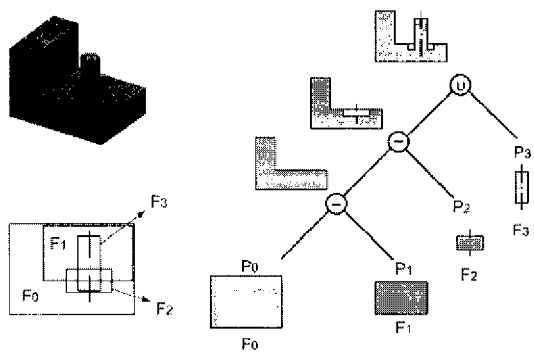


Fig. 2 A feature modeling tree

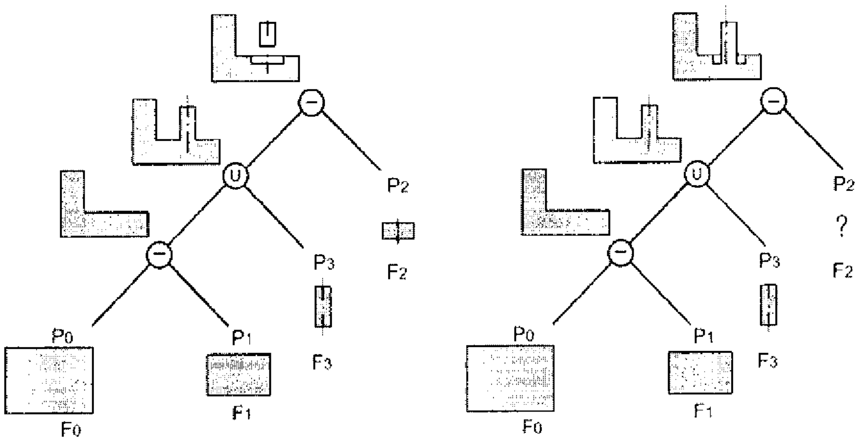


Fig. 3 Reordering of the Boolean operations

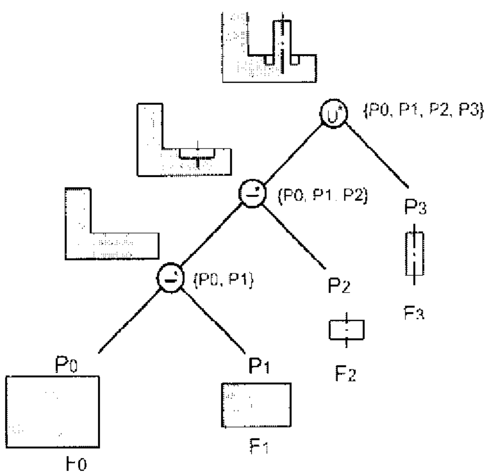


Fig. 4 A CSG tree of the history-based Boolean operations for the example shown in Fig. 2

regular geometric arrangement, such as circles or array patterns.

A part-modeling procedure can be represented by a feature-modeling tree as shown in Fig. 2. In this tree, the leaf nodes represent the primitives of the features, and the intermediate nodes represent the Boolean operations, which contain either a union or subtraction operation. To build the tree, it is necessary to convert transition and pattern features into volume features, and to reclassify them as additive or subtractive.

For multi-resolution modeling, the features need to be rearranged in proportion to the significance of the feature. However, if features are rearranged, the resulting shape can be different from the original, because mixed Boolean operations of union and subtraction do not obey a commutative law. For example, if the features in Fig. 2 are rearranged to $F_0 \rightarrow F_1 \rightarrow F_3 \rightarrow F_2$, as shown in Fig. 3(a), the result differs from the original shown in Fig. 2. Fig. 3(b) represents what it should be : the highest LOD model must be the same shape as the original in spite of an arbitrary rearrangement of features, and the intermediate LOD models must have reasonable shapes.

3. Definition of History-Based Boolean Operations

When the order of Boolean operations is changed, the region affected by each Boolean operation in the initial order can be different from that in the rearranged order. This makes the union and subtraction operations non-commutative. Our idea is to utilize the modeling history to make these operations commutative. The primitives used in each operation are first stored and then used to provide the same result after the reordering of the Boolean operations. In this paper, the Boolean operations that obey the commutative law between union and subtraction operations by using the modeling history are called history-based Boolean operations, and the primitives stored with each operation are called affecting primitives Note that intersection operations are not included in the history-based Boolean operations because feature-based modeling is implemented using only union and subtraction operations The formal definition of the history-based Boolean operation is as follows

[Definition 1] History-based Boolean operations

Let U, P, and Q denote the sets of primitives as follows U is a set of all primitives that participate in the Boolean operations for modeling a part; P is a subset of U that includes the primitives in the child nodes of a given operation node, and Q is a subset of U such that $Q=U-P$, that is $Q=\{Q_i \mid Q_i \notin P \wedge Q_i \in U\}$ where Q_i denotes a primitive in Q That is, Q is a set of the primitives used in the subsequent Boolean operations, whereas P is a set of the primitives used in the previous Boolean operations Then, if the history-based union and subtraction Boolean operations, which are denoted by $\cup^*_P$ and $-^*_P$ respectively, are defined as

$$A \cup^*_P B = A \cup (B - \bigcup_i Q_i) \qquad (1)$$

$$A -^*_P B = A - (B - \bigcup_i Q_i) \qquad (2)$$

where $\bigcup_i Q_i$ represents the union of all the primitives Q_i in Q

Eqs (1) and (2) also can be written as

$$A \cup^*_P B = \{x \mid x \in A \vee (x \in B \wedge x \notin Q_i)\} \qquad (3)$$

$$A -^*_P B = \{x \mid x \in A \vee \neg(x \in B \wedge x \notin Q_i)\} \qquad (4)$$

where $\vee$, $\wedge$, and $\neg$ symbolize 'and', 'or', and 'not' respectively, and x denotes a point in the 3-D Euclidian space Fig 4 illustrates the CSG tree of the history-based Boolean operations for the example in Fig 2 Here, $\{P_i\}$ represents the set P that consists of the primitives of the previous Boolean operations

4. Commutative Property of History-Based Boolean Operations

If two operations are selected from union and subtraction, there are four combinations $\cup$ and $\cup$, $-$, and $-$, $\cup$ and $-$, and $-$ and $\cup$ In the following sections, the commutative property of history-based Boolean operations is investigated for three cases two unions, two subtractions, and a union and a subtraction The proof of each equation is described in detail in the Appendix

4.1 Commutative property of history-based union operations

It is well known that union operations are commutative $A \cup B \cup C = A \cup C \cup B$ (Lin, 1974) We investigate whether history-based union operations are also commutative The modeling process $A \cup B \cup C$ is represented by $A \cup^*_{\{A,B\}} B \cup^*_{\{A,B,C\}} C$ in history-based Boolean operations From Eq (1) and the laws of Boolean algebra (Lin, 1974),

$$\begin{aligned} A \cup^*_{\{A,B\}} B \cup^*_{\{A,B,C\}} C &= A \cup (B - C) \cup (C) \\ &= A \cup B \cup C \end{aligned} \qquad (5)$$

$$\begin{aligned} A \cup^*_{\{A,B,C\}} C \cup^*_{\{A,B\}} B &= A \cup C \cup (B - C) \\ &= A \cup B \cup C \end{aligned} \qquad (6)$$

As $A \cup^*_{\{A,B\}} B \cup^*_{\{A,B,C\}} C = A \cup^*_{\{A,B,C\}} C \cup^*_{\{A,B\}} B$ history-based union operations are commutative The detailed derivation is given in the Appendix

4.2 Commutative property of history-based subtraction operations

It is also known that subtraction operations are commutative $A - B - C = A - C - B$ In history-based Boolean operations, the modeling process $A - B - C$ is represented by $A -^*_{\{A,B\}} B -^*_{\{A,B,C\}} C$ From Eq (1) and the laws of Boolean algebra,

$$\begin{aligned} A -^*_{\{A,B\}} B -^*_{\{A,B,C\}} C &= A - (B - C) - C \\ &= A - B - C \end{aligned} \qquad (7)$$

$$\begin{aligned} A -^*_{\{A,B,C\}} C -^*_{\{A,B\}} B &= A - C - (B - C) \\ &= A - B - C \end{aligned} \qquad (8)$$

Because $A -^*_{\{A,B\}} B -^*_{\{A,B,C\}} C = A -^*_{\{A,B,C\}} C -^*_{\{A,B\}} B$, history-based subtraction operations are commutative

4.3 Commutative property of history-based union and subtraction operations

First, consider the case of $A \cup B - C$ In conventional Boolean operations, the resultant shape

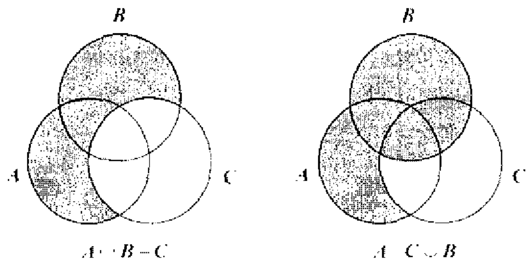


Fig. 5 Venn diagrams of $A \cup B - C$ and $A - C \cup B$

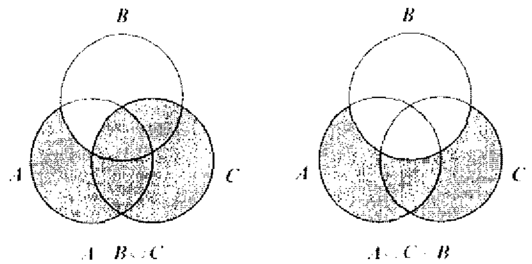


Fig. 6 Venn diagrams of $A - B \cup C$ and $A \cup C - B$

of $A \cup B - C$ is different from that of $A - C \cup B$ as shown in Fig. 5. However, in history-based Boolean operations

$$A \cup^*_{\{A,B\}} B -^*_{\{A,B,C\}} C = A \cup (B - C) - C = A \cup B - C \tag{9}$$

$$A -^*_{\{A,B,C\}} C \cup^*_{\{A,B\}} B = A - C \cup (B - C) = A \cup B - C \tag{10}$$

Eqs. (9) and (10) show that the reordered history-based Boolean operations provide the same result as the original.

Next, consider the case of $A - B \cup C$. As illustrated in Fig. 6, for conventional Boolean operations, the resultant shape of $A - B \cup C$ is different from that of $A \cup C - B$. However, for history-based Boolean operations,

$$A -^*_{\{A,B\}} B \cup^*_{\{A,B,C\}} C = A - (B - C) \cup C = A - B \cup C \tag{11}$$

$$A \cup^*_{\{A,B,C\}} C -^*_{\{A,B\}} B = A \cup C - (B - C) = A - B \cup C \tag{12}$$

Eq. (12) shows that the result of reordering the history-based Boolean operations is the same as the original.

From the results in the two cases $A \cup B - C$ and $A - B \cup C$, it follows that mixed history-based Boolean operations of union and subtraction are commutative.

5. Extension of Affecting Primitives

Although history-based Boolean operations guarantee the same resultant shape for an arbitrary reordering of the operations, solid models at the intermediate LODs may have unnatural shapes. For example, if the Boolean sequence in Fig. 2 is reordered to $F_0 \rightarrow F_1 \rightarrow F_3 \rightarrow F_2$, the intermediate LOD models will be as shown in Fig. 7. For each Boolean operation, the volume originating only from the affecting primitives of the operation is used as a tool body. For example, in the first subtraction operation $P_0 -^*_{\{P0,\ P1\}} P_1$, the volume overlapping P_2 and P_3 is excluded from P_1. However, in this case, an undesirable detailed shape appears at the LOD=1. This shape should be eliminated to provide a more natural LOD model. Consequently, in order for the history-based Boolean operation to be applicable to feature-based multi-resolution modeling, it is essential to develop an algorithm to provide more reasonable solid models at intermediate LODs.

These unacceptable intermediate LOD models originate from the algorithm that excludes in advance the region overlapping with primitives used by subsequent Boolean operations. To prevent this, we extend the range of affecting primitives without violating the commutative laws of history-based Boolean operations. The affecting primitive list of a Boolean operation includes the tool bodies of the following Boolean operations. Although these primitives are included in the affecting primitive list, the final shape is not changed because the region overlapping with these primitives will be modified later by their corresponding Boolean operations. For example, by adding P_2 and P_3 to the affecting primitive list of the first subtraction operation in the CSG tree in Fig. 7, the first LOD model has a more reasonable shape as shown in Fig. 8. Naturally, the highest LOD model has the same shape as

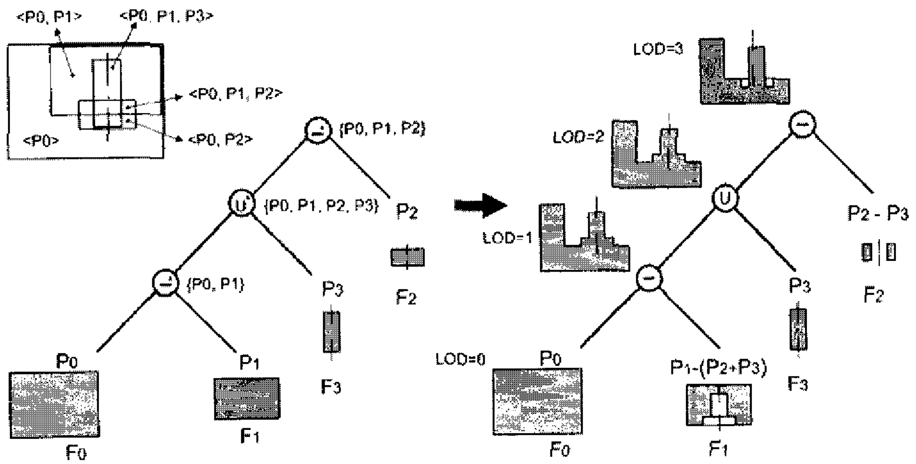


Fig. 7 A CSG tree for the rearranged history-based Boolean operations for the example shownin Fig. 3

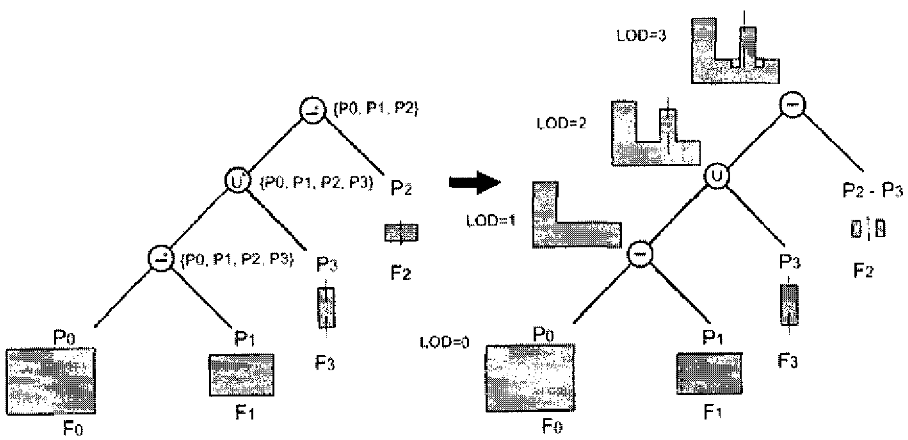


Fig. 8 Extension of affecting primitives for the example shown in Fig. 7

the original. A sequence of history-based Boolean operations gives exactly the same result as the conventional Boolean operations ordered in the original sequence. For example, if the initial modeling is $P_0 - P_1 \cup P_2 - P_3$, then $P_0 -^* P_3 = P_0 - P_3$, $P_0 -^* P_3 -^* P_1 = P_0 - P_1 - P_3$, $P_0 -^* P_3 \cup^* P_2 -^* P_1 = P_0 - P_1 \cup P_2 - P_3$, etc. Consequently, the history-based Boolean operations have the effect of rearranging the reordered operations to be in the initial order.

6. Conclusion

We propose history-based Boolean operations that satisfy the commutative laws for union and subtraction operations, and have developed an algorithm for multi-resolution modeling based on the non-manifold merged set and history-based Boolean operations. This algorithm guarantees the same resultant shape and reasonable intermediate LOD models for an arbitrary rearrangement of the features, consistent with a certain LOD criterion, such as the volume of the feature, regardless of whether the feature is subtractive or additive.

As future work, the challenge is to extend the multi-resolution modeling technique to multi-abstraction modeling that can provide geometric models at various levels of abstraction for engineering analysis. To accomplish this goal, it is necessary to extend the representation domain of history-based Boolean operations from solid to non-manifold models

Acknowledgment

This work was supported by grant No R01-2002-000-00061-0 from the Basic Research Program of the Korea Science & Engineering Foundation

References

Cignoni, P, Montani, C and Scopigno, R, 1998, "A Comparison of Mesh Simplification Algorithms," *Computers & Graphics*, Vol 22 No 1, pp 37~54

Choi, D H, Kim, T W and Lee K, 2002, "Multiresolutional Representation of B-rep Model Using Feature Conversion," *Transactions of the Society of CAD/CAM Engineers*, Vol 7, No 2, pp 121~130 (in Korean)

Crocker, G A and Reinke, W F, 1991, "An Editable Nnon-Manifold Boundary Representation," *IEEE Computer Graphics & Applications*, Vol 11, No 2, pp 39~51

Dunn, M, 1992, Industrial Automation Systems and Integration — Product Data Representation and Exchange — Part 48 integration Generic Resources Form Features, Second Edition ISO/WD 10303-48

Kim, S C, Lee, K and Kim, T W, 2003, "Feature Tree Pruning Operation to Generate Multi-Resolution Model," *Proceedings of 2003 Conference of the Society of CAD/CAM Engineers*, Seoul, Korea, January 30, 2003, pp 397~402 (in Korean)

Kim, S H, Lee, K and Kim, Y -J, 1996, "Development of Boolean Operations for CAD System Kernel Supporting Non-Manifold Models," *Transactions of the Society of CAD/CAM Engineers*, Vol. 1, No. 1, pp 20~32.(in Korean)

Koo, S and Lee, K, 2002, "Wrap-Around Operation to Make Multi-Resolution Model of Part and Aassembly," *Computers & Graphics*, Vol 26, No 5, pp 687~700

Lee, J Y, Lee, J -H, Kim, H. and Kim, H S, 2002, "Progressive Solid Models for Internet-based Design and Collaboration," *Proceedings of the 2002 ASME Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, Montreal, Canada, September 29 — October 2, 2002, DETC2002/CIE-34457

Lee, J Y, Lee, J -H, Kim, H and Kim, H S, 2004, "A Cellular Topology-based Approach to Generating Progressive Solid Models from Feature-centric Models," *Computer-Aided Design*, Vol 36, No 3, pp 217~229

Lee, K, 1999, *Principles of CAD/CAM/CAE Systems*, Addison Wesley Longman, Inc

Lee, S H, Lee, K -S and Park, S, 2002, "Multiresolution Representation of Solid Models Using the Selective Boolean Operations," *Proceedings of 2002 Spring Conference of Korean Society of Precision Engineering*, Daejeon, Korea, May 17-18, 2002, KSPE 02S179, pp 833~835 (in Korean)

Lin, S -Y T and Lin, Y -F, 1974, *Set Theory · An Intuitive Approach*, Boston, Houghton Mifflin

Masuda, H, 1992, "Topological Operators and Boolean Operations for Complex-based Non-manifold Geometric Models," *Computer-Aided Design*, Vol 25, No 2, pp 119~129

Schroder, W J, Zerge, J A and Lorensen, W E, 1992, "Decimation of Triangle Meshes," *Proceedings of SIGGRAPH '92 in Computer Graphics*, Vol 26, No 2, pp 65~70

Shah, J J and Mantyla, M, 1995, "Parametric and Feature-based CAD/CAM," *John Wiley & Sons*, Inc

Appendix

The detailed proofs of Eq (5) to (12) are described in this section If A^c denotes the complement of A, the laws of Boolean algebra are summarized into Table 1 The subtraction operation can be represented by (Lin, 1974)

$$A-B=A\cap B^c \qquad (13)$$

Table 1 Laws (axioms) of boolean algebra

Name	Axiom	Dual
Idempotent	$A\cup A=A$	$A\cap A=A$
Identity	$A\cup\phi=A$ $A\cup U=U$	$A\cap U=A$ $A\cap\phi=\phi$
Complement	$A\cup A^c=U$ $(A^c)^c=A$	$A\cup A^c=\phi$ $(A^c)^c=A$
Commutative	$A\cup A=B\cup A$	$A\cap B=B\cap A$
Associative	$(A\cup B)\cup C$ $=A\cup(B\cup C)$	$(A\cap B)\cap C$ $=A\cap(B\cap C)$
Distributive	$A\cap(B\cup C)$ $=(A\cap B)\cup(A\cap C)$	$A\cup(B\cap C)$ $=(A\cup B)\cap(A\cup C)$
De Morgan's	$(A\cup B)^c=A^c\cap B^c$	$(A\cap B)^c=A^c\cup B^c$
Absorption	$A\cup(A\cap B)=B$ $A\cup(A^c\cap B)=A\cup B$	$A\cap(A\cup B)=A$ $A\cap(A^c\cup B)=A\cap B$

A.1 Proof of Eq. (5).

$$A\cup^*_{\{A,B\}}B\cup^*_{\{A,B,C\}}C=A\cup B\cup C$$

$A\cup^*_{\{A,B\}}B\cup^*_{\{A,B,C\}}C$

$=A\cup(B-C)\cup(C)$ (Eq (1) and Eq (2))

$=A\cup(B\cap C^c)\cup C$ (Eq (13))

$=A\cup((B\cap C^c)\cup C)$ (associative law)

$=A\cup((B\cup C)\cap(C^c\cup C))$ (distributive law)

$=A\cup((B\cup C)\cap U)$ (complement law)

$=A\cup(B\cup C)$ (identity law)

$=A\cup B\cup C$ (associative law)

A.2 Proof of Eq. (6)·

$$A\cup^*_{\{A,B,C\}}C\cup^*_{\{A,B\}}B=A\cup B\cup C$$

$A\cup^*_{\{A,B,C\}}C\cup^*_{\{A,B\}}B$

$=(A\cup C)\cup(B-C)$ (Eq (1) and Eq (2))

$=(A\cup C)\cup(B\cap C^c)$ (Eq (13))

$=((A\cup C)\cup B)\cap((A\cup C)\cup C^c)$ (distributive law)

$=((A\cup B)\cup C)\cap((A\cup C)\cup C^c)$ (commutative law)

$=((A\cup B)\cup C)\cap(A\cup(C\cup C^c))$ (associative law)

$=((A\cup B\cup C)\cap(A\cup U))$ (complement law)

$=(A\cup B\cup C)\cap U$ (identity law)

$=A\cup B\cup C$

A.3 Proof of Eq. (7)

$$A-^*_{\{A,B\}}B-^*_{\{A,B,C\}}C=A-B-C$$

$A-^*_{\{A,B\}}B-^*_{\{A,B,C\}}C$

$=A-(B-C)-C$ (Eq (1) and Eq (2))

$=(A\cap(B\cap C^c)^c)\cap C^c$ (Eq (13))

$=(A\cap(B^c\cup C))\cap C^c$ (De Morgan's law)

$=((A\cap B^c)\cup(A\cap C))\cap C^c$ (distributive law)

$=((A\cap B^c)\cap C^c)\cup((A\cap C)\cap C^c)$ (distributive law)

$=((A\cap B^c)\cap C^c)\cup(A\cap(C\cap C^c))$ (associative law)

$=((A\cap B^c)\cap C^c)\cup(A\cap\phi)$ (complement law)

$=((A\cap B^c)\cap C^c)\cup\phi$ (identity law)

$=A-B-C$ (Eq (13))

A.4 Proof of Eq. (8).

$$A-^*_{\{A,B,C\}}C-^*_{\{A,B\}}B=A-B-C$$

$A-^*_{\{A,B,C\}}C-^*_{\{A,B\}}B$

$=(A-C)-(B-C)$ (Eq (1) and Eq (2))

$=(A\cap C^c)\cap(B\cap C^c)^c$ (Eq (13))

$=(A\cap C^c)\cap(B^c\cup C)$ (De Morgan's law)

$=((A\cap C^c)\cap B^c)\cup((A\cap C^c)\cap C)$ (distributive law)

$=((A\cap C^c)\cap B^c)\cup(A\cap(C^c\cap C))$ (associative law)

$=((A\cap C^c)\cap B^c)\cup(A\cap\phi)$ (complement law)

$=((A\cap C^c)\cap B^c)\cup\phi$ (identity law)

$=(A\cap B^c)\cap C^c$ (commutative law)

$=A-B-C$ (Eq (13))

A.5 Proof of Eq. (9).

$$A\cup^*_{\{A,B\}}B-^*_{\{A,B,C\}}C=A\cup B-C$$

$A\cup^*_{\{A,B\}}B-^*_{\{A,B,C\}}C$

$=A\cup(B-C)-(C)$ (Eq (1) and Eq (2))

$=A\cup(B\cap C^c)\cap C^c$ (Eq (13))

$=(A\cup B)\cap(A\cup C^c)\cap C^c$ (distributive law)

$=(A\cup B\cap((A\cup C^c)\cap C^c)$ (associative law)

$=(A\cup B)\cap C^c$ (absorption law)

$=A\cup B-C$ (Eq (13))

A.6 Proof of Eq. (10)

$$A-^*_{\{A,B,C\}}C\cup^*_{\{A,B\}}B=A\cup B-C$$

$A-^*_{\{A,B,C\}}C\cup^*_{\{A,B\}}B$

$=A-C\cup(B-C)$ (Eq (1) and Eq (2))

$=(A\cap C^c)\cup(B\cap C^c)$ (Eq (13))

$=((A\cap C^c)\cup B)\cap((A\cap C^c)\cup C^c)$ (distributive law)

$=((A\cap C^c)\cup B)\cap C^c$ (absorption law)

$=((A\cup B)\cap(C^c\cup B))\cap C^c$ (distributive law)

$=(A\cup B)\cap((C^c\cup B)\cap C^c)$ (associative law)

$=(A\cup B)\cap C^c$ (absorption law)

$=A\cup B-C$ (Eq (13))

A.7 Proof of Eq. (11).

$$A -^{*}_{\{A,B\}} B \cup^{*}_{\{A,B,C\}} C = A - B \cup C$$

$A -^{*}_{\{A,B\}} B \cup^{*}_{\{A,B,C\}} C$

$= A - (B - C) \cup C$	(Eq (1) and Eq (2))
$= A \cap (B \cap C^c)^c \cup C$	(Eq (13))
$= A \cap (B^c \cup C) \cup C$	(De Morgan's law)
$= (A \cap B^c) \cup (A \cap C) \cup C$	(distributive law)
$= (A \cap B^c) \cup ((A \cap C) \cup C)$	(associative law)
$= (A \cap B^c) \cup C$	(absorption law)
$= A - B \cup C$	(Eq (13))

A.8 Proof of Eq. (12).

$$A \cup^{*}_{\{A,B,C\}} C -^{*}_{\{A,B\}} B = A - B \cup C$$

$A \cup^{*}_{\{A,B,C\}} C -^{*}_{\{A,B\}} B$

$= A \cup C - (B - C)$	(Eq (1) and Eq (2))
$= (A \cup C) \cap (B \cap C^c)^c$	(Eq (13))
$= (A \cup C) \cap (B^c \cup C)$	(DeMorgan's law)
$= (A \cap B^c) \cup C$	(distributive law)
$= A - B \cup C$	(Eq (13))