A FRAMEWORK IN DEVELOPING KERNEL MODELER BASED ON NONMANIFOLD BOUNDARY REPRESENTATION

Younghyun Han, Kunwoo Lee, Youngjin Kim, Gangsoo Lee,
Jinwoong Hong, Jaehong Ahn, Junghoon Hur, Kyoungjin Lee
Department of Mechanical Design and Production Engineering
Seoul National University
Seoul, Korea

Sanghun Lee
Institute for advanced Engineering, Daewoo

Sunghwan Kim
Daeyeu Technical Junior College

ABSTRACT

Nonmanifold representation of models has been winning a great popularity because of its superiority to conventional manifold representations in representing the complicated topology of an object and its suitability to various applications that have been difficult to be solved by manifold models. The paper describes an geometric kernel modeler which is based on a boundary representation for nonmanifold models. The topological representation scheme used in the kernel system is basically a familiar hierarchical structure between topological elements, which can represent mixed dimensional models and cellular models in a unified structure. There are two types of topological elements in this scheme; one is the primitive topological elements like region, face, edge and vertex, the other is a kind of virtual elements called *partial topological elements*. The partial topological elements efficiently represent the adjacency relationship between primitive elements without redundancy of topological information that has been a drawback of the existing data structures for non-manifold models. Along with the data structure, we also present an Euler formula and a set of Euler operators that can be applied to the data structure, to guarantee the topological integrity of the modeling operation. The modeling system has been developed as a kernel system under the object-oriented frame, making itself serve as a flexible and extensible geometric engine for various applications. The high-level modeling operations like sweeping, Boolean operation are developed to facilitate constructing complex objects, including nonmanifold models. These operations provided in the kernel system are also illustrated with some modeling examples.

INTRODUCTION

Geometric modeling systems have been widely used in industry for the design and manufacture of products. However, the conventional geometric modeling systems have several problems in providing an integrated environment for the product development process as follows. First, they have limitations in representing different models required in various stages of the product development; conceptual design, final design, analysis and manufacture. As a solution to this, geometric modeling system supporting nonmanifold topology has been suggested. The nonmanifold geometric modeling system provides a unified topological representation of wireframe, surface, solid models and mixed dimensional objects, and make themselves a basic tool for the integration of the product development process such as design, analysis and manufacture of a product. Another problems is the closed structure of the conventional modeling systems. That is, users of the systems can only utilize the functions provided by the system; they cannot get any benefit in the right time from the system if it does not support or provide a certain function specific to the application areas of the users. Though some systems have tried to solve this problem by providing interface subroutines which allow user to access to the internal structure and modeling functions of the system, it does not suffice to meet all of the requirements from the user. Recently, the kernel modeler has shown up to solve this problem. In a kernel modeler, by disclosing its topological and geometrical data structure as well as providing interface routines manipulating geometric models, it enables users of the system to develop their own applications appropriate to their particular purpose.

In this paper, we describe the kernel modeler based on a new and compact data structure for boundary representation of nonmanifold models. The topological data structure for nonmanifold objects in this system is described in the second section. The third section provides the Euler formula and a set of Euler operations that

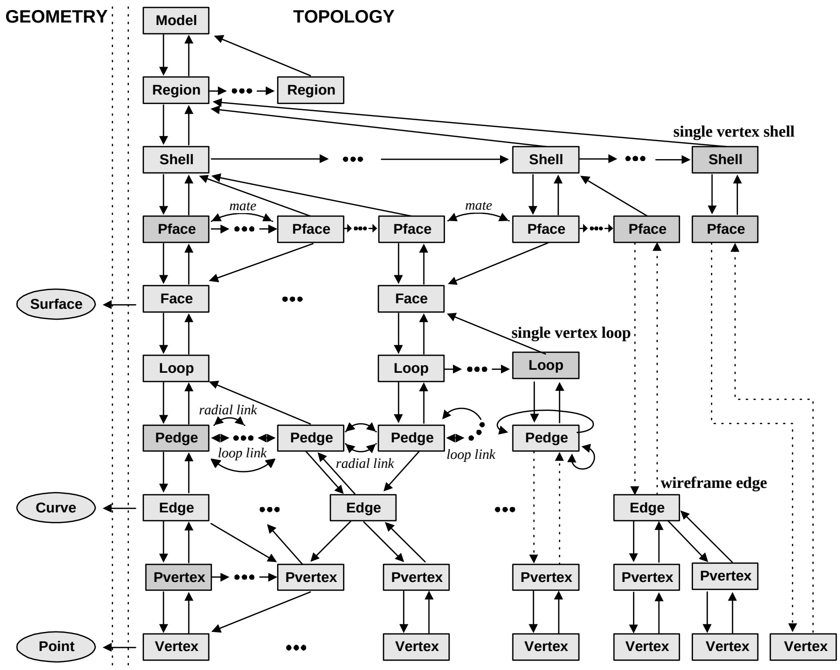


FIG. 1 TOPOLOGICAL DATA STRUCTURE FOR NONMANIFOLD MODELS

we use to guarantee the integrity of the topological structure of geometric models through the modeling operations. In fourth section, a brief description on the structure of the kernel system and the bases on which the kernel system is designed and implemented is given. The essential modeling operations such as sweeping, Boolean operation are also presented with modeling examples in the fifth section. Finally, conclusion is given in the last section.

DATA STRUCTURE FOR NONMANIFOLD TOPOLOGY

Several representation scheme for nonmanifold topology has been proposed by many researchers. Weiler(1986) first proposed *Radial-edge structure*, in which structure a new element *use* associated with face, loop, edge and vertex is introduced to represent the adjacency relationships between the topological elements. Choi(1989) proposed *Vertex-based B-rep* to deal with the nonmanifold condition around the vertex which has been handled incompletely in Weiler's work. Rossignac and O'Connor(1990) proposed *Selective geometry complex* based on an incidence graph of n-dimensional cells. Yamaguchi(1991) represented nonmanifold models by introducing *coupling entities* for adjacency relationships. Lee(1993) proposed a compact hierarchical nonmanifold boundary representation that is extended from half-edge data structure for solid models, in which scheme he introduced *partial topological elements* to represent the nonmanifold conditions around a vertex, edge or face.

In our kernel modeler, we accepted the representation scheme proposed by Lee as a topological data structure for nonmanifold models, and its schematic diagram is shown in Fig. 1. This representation scheme is a familiar hierarchical data structure of the primitive topological elements such as model, region, shell, face, loop, edge, vertex. Besides the primitive elements, the three partial topological elements(Pface, Pedge, Pvertex) are also introduced to represent the adjacency relationships between the primitive topological elements. We will briefly describe these partial topological elements in the following.

First, the partial faces lie in both sides of the face, making up parts of the shells bounding the regions in each side of the face. In terms of a face, the face is associated with two partial face(*mate* relation in Fig. 1) in both sides of it, thus with two shells and two regions, except for some special cases. With this concept, the cellular model with interior partitions can be represented in this representation scheme. In case of laminar face, the partial faces in each side of it is associated with the same shell. As exceptional cases, a single vertex shell(an isolated vertex) and a wireframe edge that are not pertaining to a specific face are also represented by directly connecting the primitive topological elements of vertex and edge to a dummy partial face, as indicated by dotted line in Fig. 1.

Second, the partial edge represents the adjacency relationship between a loop and an edge. In case of a

TAB. 1 EULER OPERATORS

	Name	Description	v	e	f	L	S	C	R
Basic Euler Operators	MEV(KEV)	make(kill) edge, vertex	1	1	0	0	0	0	0
	MEC(KEC)	make(kill) edge, cycle	0	1	0	0	0	1	0
	MFKC(KFMC)	make(kill) face, kill(make) cycle	0	0	1	0	0	-1	0
	MFR(KFR)	make(kill) face, region	0	0	1	0	0	0	1
	MVS(KVS)	make(kill) vertex, shell	1	0	0	0	1	0	0
	MVL(KVL)	make(kill) vertex, loop	1	0	0	1	0	0	0
Additional Euler Operators	SEMV(JEKV)	split(join) edge, make(kill) vertex	1	1	0	0	0	0	0
	MEF(KEF)	make(kill) edge, face	0	1	0	0	1	0	0
	KEML(MEKL)	kill(make) edge, make(kill) loop	0	-1	0	1	0	0	0
	KEMS(MEKS)	kill(make) edge, make(kill) shell	0	-1	0	0	1	0	0
Additional Topological Operators	MMR(KMR)	make(kill) model, region							

single vertex loop as shown in Fig. 1, a dummy partial edge directly connects a loop to a partial vertex. The partial edge contains two cyclic orders: the loop cycle and the radial cycle. The loop cycle is the order of edges in a loop and the *radial cycle* is the order of faces around an edge; each of which is indicated as *loop link* and *radial link* in Fig. 1. With the radial link between partial edges, the nonmanifold condition around an edge can be represented in this representation scheme. In this representation scheme, the number of partial edges around an edge is the same as the number of faces around an edge.

Third, the partial vertex, which is a unique topological entity introduced in our representation scheme, represents the topological relationship between a vertex and the 2-manifolds adjacent to the vertex. Here, the 2-manifold means a face or connected faces around a vertex. As an exceptional case, the wireframe edge is considered to be a degenerate manifold. In a manifold model, a vertex is always adjacent to a 2-manifold and thus only one partial vertex is associated with a vertex, including the case of a single vertex loop. In case of nonmanifold condition around a vertex, the number of partial vertices associated with a vertex is the same as the number of the respective 2-manifolds and the wireframe edges adjacent to the vertex.

With the above-mentioned representation scheme, our kernel modeler can handle mixed dimensional models and cellular models in a unified structure. Note that the partial topological elements just represent the adjacency relationships between the primitive topological elements and do not contain any boundary information itself, which is different from the early representation schemes by Weiler, Choi or Yamaguchi. More detailed description on this data structure and the comparison with the other schemes are given in Lee's work(1993).

EULER OPERATIONS

Most solid modeling systems use Euler operators to maintain the topological integrity of the models in implementing the high-level modeling operations. Several researchers such as Yamaguchi(1991), Masuda(1993) and Lee(1993) have derived proper Euler formula that can be applied to nonmanifold models and proposed Euler operator set which is applicable to the corresponding topological structure. In our system, we also accepted the one by Lee, together with the data structure proposed by him. We will give an brief description on the Euler formula and Euler operators used in our kernel modeler in the following.

The basic formula that can be applied to our topological data structure is as follows.

$$v - e + f - L = S - C + R \tag{1}$$

where v, e, f are the number of vertices, edges and faces, L is the number of hole loops, S is the number of void shells in regions, C is the number of cycles which can not be contractible to a point, R is the number of regions. From Eq. (1), we can see that six independent Euler operators and six inverse operators are sufficient to manipulate the topological structure of nonmanifold models in our system. For the convenience of the modeling operations, however, eight additional Euler operators are supplemented, and two topological operators are added that, though not directly related to the Euler formula, generate or delete the dummy model and region when initially creating the model or finally deleting the model. In Table. 1 are listed the basic and extended Euler operators used in our system. All the high-level modeling operations such as sweeping, Boolean operation are implemented using the twenty Euler operators in our kernel modeler.

KERNEL SYSTEM

ACIS(1994) is the most widespread kernel system in the world and many other successful CAD/CAM packages have been utilized ACIS as a geometric engine. Besides ACIS, in recent years, several kernel modeling systems supporting nonmanifold topology have been developed and can be available as a commercial package. It is commonly recognized that these kernel systems help to develop a variety of applications that require manipulation of geometric entities, and to make them to be integrated with each other without interface problems. Our kernel modeler has been developed as a part of the project of *Advanced Manufacturing System*. There are so many different applications like CAD/CAM/CAE in that system and each of them requires a variety of modeling capabilities for different class of models. Thus, a geometric kernel system with open architecture was necessary to meet all the different modeling requirement of various applications in that system and to make them cooperate in a unified environment. This was the motivation of developing this kernel modeler that could support various geometric models such as nonmanifold models in a unified data structure, for the purpose of serving as a geometric modeling kernel of various applications in that system.

The structure of the kernel system can be categorized into geometric engine, topological engine, modeling engine, APIs and supporting module. The geometric engine includes NURBS curve/surface manipulation module, intersection module and basic geometric calculation module. The topological engine comprises topology manipulation module and Euler operation module. High-level modeling operations like generation of primitive solids, local modification, sweeping and Boolean operation are included in the modeling engine. The supporting module provides the supplementary modeling facilities such as undo/redo, journaling and error recovery. The APIs are a collection of interface routines to the kernel functionality, by which the users can access and utilize the modeling operations and facilities conveniently.

This kernel modeler is designed and implemented under the object-oriented frame using C++ language. The geometric and the topological entities are defined hierarchically using class derivation and inheritance mechanism of C++. Thus, all the information on the geometric or topological structures of the kernel, and all the operations on them are provided as a form of class libraries of C++ language. Therefore, all applications of this kernel system can access the kernel directly through the class libraries as well as through APIs. In addition, the applications can inherit selected classes of the geometric or topological entities provided by this kernel and define their own entity classes appropriate to their particular purpose. This open architecture of the kernel system enables particular applications to be more flexible and extensible by making its all capabilities fully available to them.

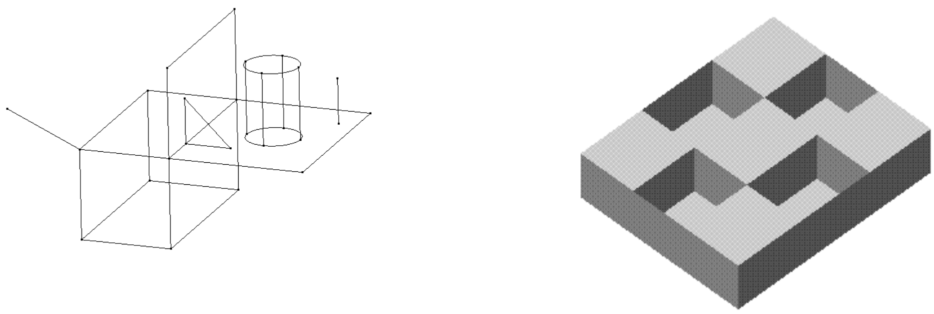
MODELING OPERATIONS

In this section, we will describe important modeling operations provided by our kernel modeler and illustrate modeling capabilities with some examples.

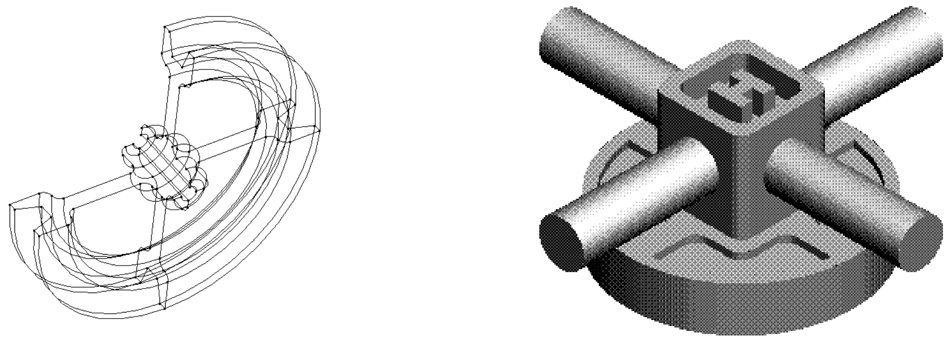
A variety of primitive solids are provided such as block, sphere, cylinder, prism, cone, pyramid, torus, circular tube, rectangular tube and frustum. Besides, diverse creation operations of the wireframe and the sheet models are also provided. From these wireframe, surface and primitive solid models could be obtained more complex models using powerful high-level modeling operations like sweeping, Boolean operation.

We can obtain one-dimension higher entities by sweeping various geometric entities; from vertex to wireframe edge or wireframe model, from edge or wireframe model to face or sheet model, from face or sheet model to solid. If a loop in a solid that consists of closed edges is swept, a sheet model without volume is added to the original solid, which results in a mixed dimensional model. The geometric entities(point, curve or surface) can also be directly swept to corresponding models(wireframe, sheet or solid models). In Fig. 2 are shown some modeling examples by sweeping operation. Two typical nonmanifold models supported in our kernel modeler are shown in (a) and two solids generated by sweeping are in (b)..

Boolean operation is a familiar and convenient modeling operation to create a nonmanifold model of complex shapes as it is in dealing with 2-manifold models. The boundary evaluation method to perform the Boolean operation in our modeling kernel is based on the concept of *Merge and Selection*, which is common to previous researches such as Gursoz(1991), Crocker(1991), and Masuda(1992).The process of Merge and Selection is started with obtaining the merged set into which all the input primitives are embedded. Thus the merged set has complete information on the applied primitives, all the intersection entities between them, and historical information why and how the elements are created. After the merged set is obtained, selection process follows in which the entities that must remain as a result of applying given CSG tree are marked as *alive*. The advantage of this approach is to enable the CSG tree to be modified easily by repeating only the selection process. And the geometric change of a primitive applied can also be reflected more effectively by extracting the primitive to be changed then adding a new primitive to the merged set. Since the merged set contains all the entities and history records in terms of given primitives, a marked merged set can includes the hybrid representation of B-rep and CSG tree in a unified nonmanifold model data structure. And it is expected that the merged set would be a fundamental tool in many modeling problems such as data representation of form features, and the interference management between features, and data representation of conceptual models in design process, etc. An example is shown in Fig. 3. The merged set in (a) is built with 11 primitives, and (b), (c) are marked result with different CSG trees.

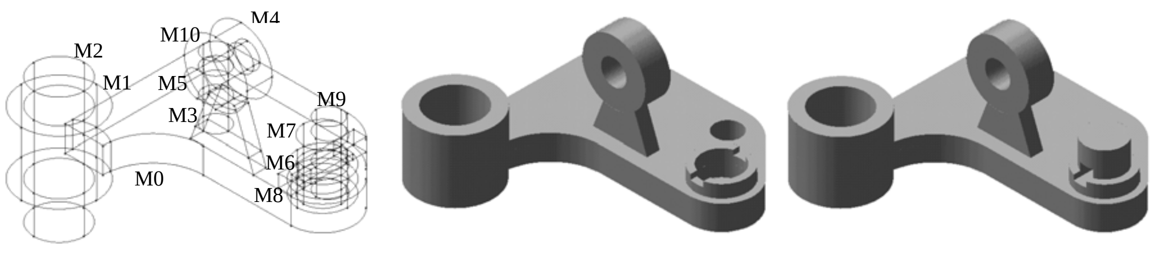


(a) Nonmanifold models: a mixed dimensional model and a cellular model



(b) Solid models by a rotational and a translational sweeping

FIG. 2 EXAMPLES OF SWEEPING OPERATION



(a) merged set

(b) CSG tree : M0+M1-M2+M3+ M4-M5+M6-M7-M8-M9-M10

(c) CSG tree : M0+M1-M2+M3+ M4-M5+M6+M7-M8

FIG.3 EXAMPLES OF BOOLEAN OPERATION

CONCLUSION

The nonmanifold modeling systems are expected to play an important role in providing integrated environment for the development of a product by providing unified modeling capabilities required in various stage of design, analysis and manufacture of a product. In addition, the modeling systems, provided as a kernel, could serve as a foundation for particular applications, helping make those applications more flexible and extensible.

Our kernel modeler is based on a compact and unified hierarchical boundary representation for nonmanifold topology, and can support nonmanifold geometric models having practical meaning in engineering purpose. It can represent wireframe, surface and solid model, or mixed dimensional models of them, and cellular models with internal faces. To facilitate modeling of an object with complex shape, high-level modeling operations such as sweeping, Boolean operation are also provided with the full utilization of the nonmanifold characteristics. Besides, all the modeling operations which change the topology of the model are implemented using Euler operators, thus guaranteeing the topological consistency of the models in our system. As a kernel modeler with open architecture, the modeling operations are provided in APIs, and all the topological or geometric information of the model can be accessed directly and freely through the C++ class libraries.

Further work is under way to bring this kernel system to completion. Rounding operation is under development and various modeling operations and facilities such as local modification, analysis of the properties of the model, interrogation functions are also being implemented at present.

ACKNOWLEDGMENT

The work described in this paper has been carried out as a part of the project, Advanced Manufacturing System(G7), and supported by Ministry of Trade, Industry and Energy, ROK.

REFERENCES

Choi, Y.,(1989), "Vertex-Based Boundary Representation of Non-manifold Geometric Models," Ph.D. Thesis, Carnegie Mellon Univ.

Crocker, G. A. and Reinke, W. F.,(1991), “An Editable Nonmanifold Boundary Representation,” IEEE Computer Graphics & Applications, March

Gursoz, E. L., Choi, Y. and Prinz, F. B.,(1991), “Boolean Set Operations on Non-manifold Boundary Representation Objects,” Computer-Aided Design, Vol. 23, No. 1

Masuda, H. et al.,(1990), "A Mathematical Theory and Applications of Non-manifold Geometric Modeling," Geometric Modeling for Engineering Applications, Editors:F.L.krause and H.Jansen, North-Holland, pp. 89-103

Masuda, H.,(1992), “Form-Feature Representation Based On Non-manifold Geometric Modeling,” Proceedings of MICAD’92

Masuda, H.,(1993), “Topological Operators and Boolean Operations for Complex-based Nonmanifold Geometric Models,” Computer-Aided Design, Vol. 25, No. 2, pp. 119-129

Lee, S.,(1993), “Feature-based Non-Manifold Geometric Modeling System to Provide Integrated Environment for Design and Analysis of Injection Molding Products,” Ph.D. Thesis, Seoul National University

Rossignac, J. and O'Connor, M.A.,(1990), "SGC:A Dimensional-Independent Model for Pointsets with Internal Structures and Incomplete Boundaries," Geometric Modeling for Product Engineering, North-Holland, pp.145-180

Spatial Technology,(1994), “ACIS Geometric Modeler 1.6 - Application Guide,” Spatial Technology

Weiler, K.,(1986), "Topological Structures for Geometric Modeling," Ph.D. Thesis, Rensselaer Polytechnic Institute

Yamaguchi, Y., Kobayashi, K., and Kimura, F.,(1991), "Geometric Modeling with Generalized Topology and Geometry for Product Engineering," Product Modeling for Computer-Aided Design and Manufacturing, Editors:J.Turner, J.pengna and M.Wozny, Elsevier Science Publishers B.V., North-Holland, pp.97-115