A GRAPH MODEL FOR FACE-TO-FACE ASSEMBLY

L. De Floriani
Institute for Applied Mathematics, C.N.R., Genova

G. Nagy
Department of Electrical, Computer, and Systems Engineering
Rensselaer Polytechnic Institute, Troy, New York 12180

ABSTRACT

The Face-to-Face Composition (FFC) Model is a modular, boundary-based description of an object. The Assembly AND/OR graph proposed by Homem de Mello and Sanderson for task planning is an efficient data structure showing all possible assembly sequences. It can be obtained directly from the FFC model by a sequence of component-merging operations.

INTRODUCTION

The automated assembly of solid objects requires both geometric modeling tools, which trace their roots to computer graphics and computer-aided design, and task-planning tools, which are more closely related to data structures and search methods spawned by Artificial Intelligence. Further integration between these disciplines is clearly required for automated manufacture. The objective of this paper is to show how a solid modeling system that we described briefly at last year's IEEE Conference on Robotics and Automation [De Floriani 1988a] leads directly to the *Assembly AND/OR Graph* proposed by Homem de Mello and Sanderson at the same conference for the derivation of partial and complete assembly sequences [Homem de Mello 1988]. (It is ironic that one of the authors of each paper had joined the ECSE Department at RPI before the Conference, but had no communication on this topic until recently.)

We will briefly recapitulate the Face-to-Face (FFC) Model for the representation and manipulation of topological and geometric entities, including enhancements developed since last year. We will then summarize Homem de Mello's and Sanderson's proposal for assembly planning. Finally, we will show an appropriate interface between the two in terms of an AND/OR graph which contains the object geometry in a form directly usable for assembly task planning.

FACE-TO-FACE COMPOSITION MODEL

Taxonomically, the FFC Model is a modular (i.e., component-by-component), partially-evaluated boundary representation derived from the Hierarchical Face Adjacency Hypergraph [Ansaldi 1985]. It combines the ease of design of constructive solid geometry with the ease of display and surface representation of fully-evaluated boundary models [Requicha 1980]. The modular nature of the model accommodates different conceptual representations as required for functional design, process planning, and automated inspection. Most importantly for assembly, consistent alternative representations of an object are obtained by merging abutting components in different sequences. The topological integrity of individual components is assured through the use of Euler operators, while localized interference information in the form of an Interference Graph ensures the validity of complete objects. Finally, the model is independent of the application-specific user interface and of the single-component representation (such as winged-edge or symmetric data structure).

During the design phase, objects are constructed by combining components at abutting faces. As in Constructive Solid Geometry, the components may be positive or negative to allow the design of through-holes, pockets and slots, but the restriction to face-to-face combinations results in direct representation of machining and surface-finish operations (which are generally face based). Consider the object in Figure 1. The pairwise relationship of the components is represented

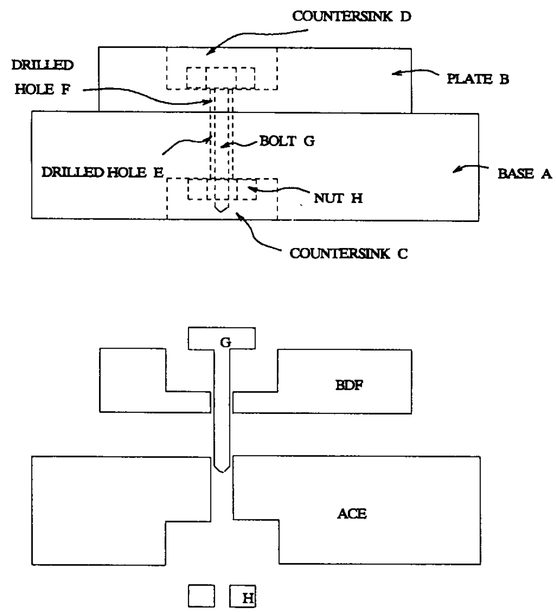

Figure 1. Base, plate, bolt, nut, countersinks, bolt-holes. The base and the plate are each the result of combining a positive object with two negative objects. The bolt is a single positive object, while the nut is made from a positive and a negative component.

CH2750-8/89/0000/0075$01.00 © 1989 IEEE

by the Connection Graph, whose nodes are valid components and whose arcs (*connection arcs*) correspond to *connection subfaces* between adjacent components. The designer specifies each face to be *real* or *virtual*. Components can be merged at virtual faces (such as the entrance to a pocket or hole) at any time during the design, but real faces demarcate the separation between components that will be manufactured separately and then assembled (Figure 2). The Connection Graph thus already contains the information required by Homem de Mello's and Sanderson's *liaison diagram*.

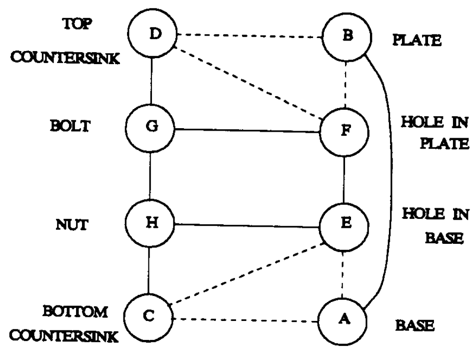

Figure 2. Connection graph for the nut-and-bolt assembly.
Virtual connection faces, which disappear in the final object, are represented by dashed arcs. Real faces are represented by solid arcs.

The relative position of two components is described in the arcs of the Connection graph in terms of the location of each connection face (i.e., the intersection of the corresponding faces of the mating components) with respect to the vertices of component faces. The Connection Graph is thus also the source of Homem de Mello's and Sanderson's *local or blocking constraints* which specify incremental motion near the final position.

The Interference Graph determines whether partially overlapping components can be merged. It is used primarily during the design phase to construct the assembly components, which are themselves complex objects obtained from simpler ones. The arc directed from component X to component Y contains all of the subfaces of Y that are properly contained inside X. The arc directed from Y to X contains the subfaces of X contained in Y. Such subfaces are called *intersection faces* (Figure 3).

Finding an acceptable merge sequence for an object can be a difficult task for objects designed as a combination of many overlapping positive and negative components. Consider the object of Figure 4. Although the final object is geometrically valid, there exists no sequence of pairwise merges for constructing it, and it is therefore not *manufacturable*. If, however, component B is split into two pieces, then the object can be constructed in the sequence A-B-C-D-E-B'. We have developed a set of heuristic rules based on the properties of the Connection and Interference graphs to find acceptable merging sequences, but in the worst case we cannot avoid a combinatorial search through the interlocking pieces [De Floriani 1988b].

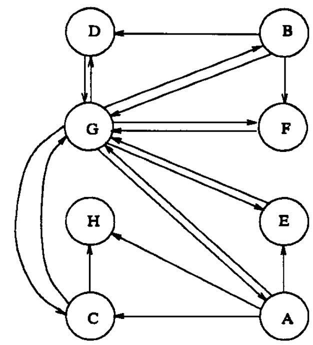

Figure 3. Interference graph for the nut-and-bold assembly.
An arc directed from A to B indicates that the volumetric component corresponding to A includes some of the faces or subfaces of B.

When two components are merged, the pairwise interference arcs (as well as the connection arcs) of the new (merged) component with all other components of the object must be recalculated. This can be accomplished systematically using the following *face algebra*.

In Figure 5, A and B are the components to be merged, AB is the merged component, and C is any other component.

Let f(A,B) denote the set of connection subfaces of A and B,

g(A,B) the set of subfaces of B included in A,

and g(B,A) the set of subfaces of A included in B.

Then the following relations can be proved when A and B are both positive components (or both negative) and their union AB is formed by merging them:

$$f(AB,C)=[f(A,C)\cup f(B,C)]-[f(A,C)\cap f(B,C)]$$

$$g(C,AB)=g(C,A)\cup g(C,B)-[g(C,A)\cap g(C,B)]$$

$$g(AB,C)=[g(A,C)\cup g(B,C)]\cup[f(A,C)\cap f(B,C)]$$

When A is positive and B is negative, and their intersection AB is formed by merging them, then the corresponding relations are:

$$f(AB,C)=[f(A,C)\cup f(B,C)]-[f(A,B)\cap f(B,C)]$$

$$g(C,AB)=[g(C,A)\cup g(C,B)]-[f(A,B)\cap g(C,B)]$$

$$g(AB,C)=g(A,C)-\{g(B,C)\cup[f(B,C)\cap g(A,C)]\}$$

The negative sign indicates the symmetric difference operation, but it can be verified that the sets subtracted are always fully contained in the sets that they are subtracted from. Examples of subfaces that are members of the various sets appearing above are shown in Figure 5. In the FFC model, the connection subfaces f(A,B) are stored in the connection arcs, while the intersection subfaces g(A,B) and g(B,A) are stored in the directed interference arcs.

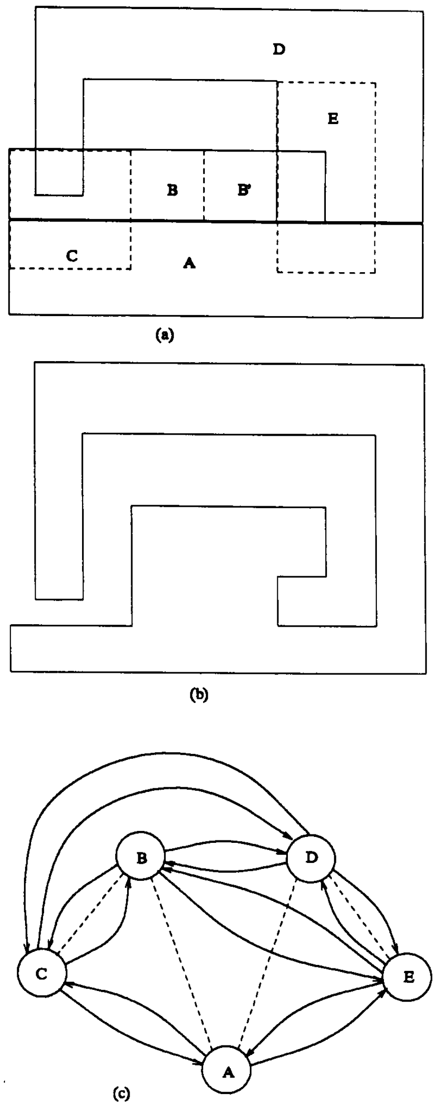

Figure 4. Valid and Invalid Objects.
The object (a) cannot be obtained by any sequence of valid pairwise merges. If, however, component B is split, then A-B-C-D-E-B' constitutes a valid sequence of merges resulting in object (b). The Connection and Interference graphs of the original object are shown in (c).

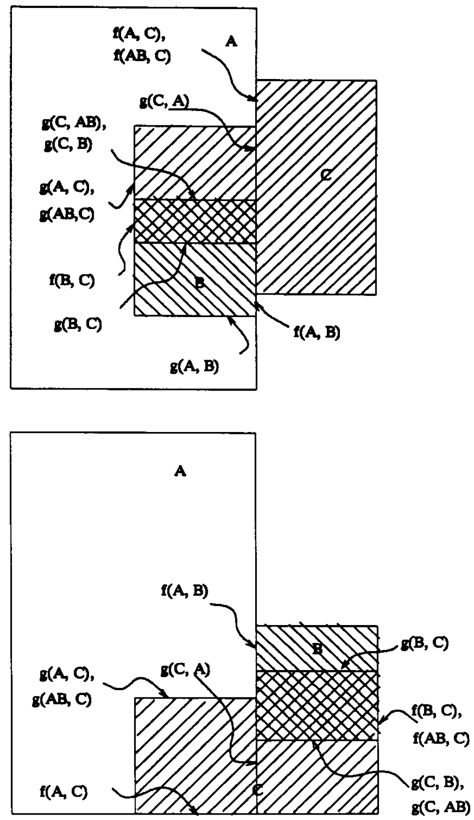

Figure 5. Subsets of faces for a face algebra.
Examples are shown of the connection and interference subfaces of objects A, B, and C. When A and B are merged, Boolean operations on these faces allow updating the arcs of the graph between the merged component AB and an arbitrary component C.

AND/OR GRAPHS

Homem de Mello and Sanderson propose searching an AND/OR graph for solution trees that represent either complete or partial assembly sequences [Homem de Mello 1986]. The partial sequences are used for the replacement of failed parts, whose identity cannot be predicted. Further, they emphasize that planning *disassembly* is easier than planning *assembly*, although each is the exact inverse of the other, because the branching factor is lower for disassembly. They point out that the AND/OR graph is an economical equivalent representation of the well-known *assembly state graph*, and provide high-level descriptions of *REPLACE-PART, SOLVE-ASSEMBLY* and *SOLVE-TASK* algorithms. Homem de Mello and Sanderson construct the AND/OR graph from cut-sets of the liaison diagram.

The best way to represent merging sequences in our model is also through the AND/OR graph. Each OR node corresponds to alternative merge sequences, while (binary) AND nodes correspond to merging two components. The initial *Design AND/OR Graph* contains all merging sequences of abutting design components: one of its subtrees is the design sequence itself. The tree is trimmed by merging all components separated by virtual faces and modifying the Connection Graph and the Interference Graph according to the face algebra. This results in the *Assembly AND/OR Graph* (Figure 6). Note that at this point all of the components are valid positive objects, hence the Interference Graph is essentially null, while the Connection Graph (Figure 7) contains all of the information necessary to compute Homem de Mello's and Sanderson's *feasiblity predicates.*

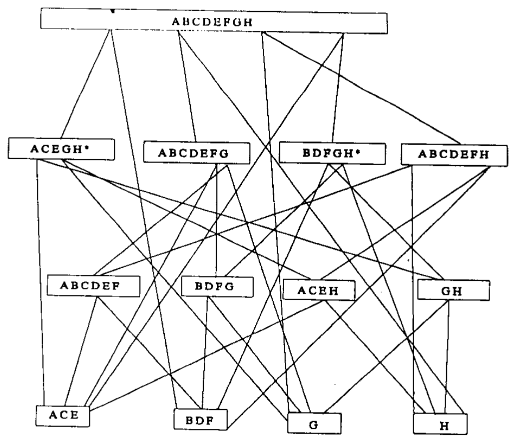

Figure 6. Assembly AND/OR graph.
This graph displays all possible face-to-face assembly sequences for the nut-and-bolt assembly of Figure 1. The boxes represent OR nodes, with AND nodes shown as the common vertices of pairs of down-links. The paths leading through the nodes marked with an asterisk are not valid, since the nut would be mounted on the bolt before either the base or the plate.

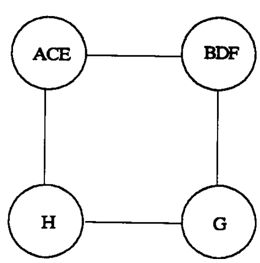

Figure 7. Assembly Connection Graph.
Thid is the result of performing the merges corresponding to every virtual connection face in Figure 1b. Here each node corresponds to a valid positive object.

At this point, the model is ready for task planning as proposed by de Mello and Sanderson. We believe that the validity of the hypothesized global trajectories necessary to position each object during assembly can be computed using the *sweep* operators which we have already implemented for simple geometries in terms of the elementary Euler operators.

ACKNOWLEDGMENT

NATO Collaborative Research Grant (# 0498/87) is gratefully acknowledged. We are indebted to Mr. A. Maulik, RPI-ECSE, for useful discussions of assembly models and for the figures.

REFERENCES

De Floriani, L. and Nagy, G., An alternative goal-oriented hierarchical representation of solid objects for computer-integrated manufacturing, *Proc. IEEE International Conference on Robotics and Automation,* Philadelphia, April 25-29, 1988, pp. 1101-1106.

Homem de Mello, L.S., and Sanderson, A., AND/OR graph representation of assembly plans, *Proc. of the Fifth National Conference on Artificial Intelligence,* 1986, pp. 1113-1119.

Ansaldi, S., De Floriani, L., Falcidieno, B., Geometric modeling of solid objects by using a face-adjacency graph representation, *Computer Graphics 19,* 3, 1985, pp. 131-139.

Requicha, A.A.G. Representation of rigid solids: Theory, methods and systems, *Computing Surveys 12,* 1980, pp. 437-464.

De Floriani, L., Maulik, A., Nagy, G. Manipulating a modular boundary model with a face-based graph structure, *Technical Report, Institute for Applied Mathematics, C.N.R., Genova, 1988.*

Homem de Mello, L.S., and Sanderson, A., Planning repair sequences using the AND/OR graph representation of assembly plans, *Proc. IEEE International Conference on Robotics and Automation,* Philadelphia, April 25-29, 1988, pp. 1861-1862.