C 23

Developments in Design for Assembly

by

S.Y. Wong* and P.K. Venuvinod**

ABSTRACT

The work described in this paper is a part of the longer effort to develop a set of algorithms to enable the computerisation of the geometric decision making involved in Design for Assembly (DFA). The paper presents a broad framework for the computer-Automated part analysis, with particular emphasis on feature recognition, for DFA. Some enhancements concerning data structures and Attributed Adjacency Graphs are proposed and discussed.

* MPhil candidate,
Department of Manufacturing Engineering,
The City Polytechnic of Hong Kong.

** Professor and Head,
Department of Manufacturing Engineering,
The City Polytechnic of Hong Kong.

1. INTRODUCTION

1.1 Assembly accounts for upto 70% of the labour cost in several discrete product industries [1]. Further, as much as 80% of the manufacturing cost of a product could be dictated by the design of the product itself [2]. Thus, rational product design for ease of assembly is of great economic significance.

1.2 The problem of product analysis for assembly planning has received much attention in recent years. In particular, a number of methods have been developed to enable the automatic generation of assembly sequences [3 to 6]. In [7], a knowledge based solution to the Design for Assembly (DFA) problem has been presented. However, perhaps the best known and most productive work in DFA is the UMass system developed by Boothroyd and Dewhurst [8]. Current applications of this system are either manual or computer-assisted. The work described in this paper is motivated by the expectation that the computerisation of the decision making involved in the UMass system should facilitate a faster assimilation of systematic DFA methods into industrial practice.

2. APPLICATIONS OF THE UMass SYSTEM

2.1 In the UMass system [8], the designer starts with an exploded view of the product, numbers the parts in the order of disassembly, and, with the aid of a series of working charts, determines the handling and insertion codes for each part. These codes are then converted into the corresponding standard handling and insertion times (or costs for automatic assembly) which are then summed up over all the parts to yield the total assembly time (or cost), TM. Each part is then critically examined to determine whether the part is essential from the point of view of the functionality, or accessibility during assembly, of the product. This process yields the theoretic minimum number of parts, NM. The 'design efficiency' of the product is then estimated as equal to 3 NM/TM which is based on the premise that the shortest possible time for the manual assembly of any pair of parts is 3 seconds.

2.2 The authors have analysed a large number of products using the UMass system and found the system to be highly systematic and effective. In [9], the system was used to assess how well Hong Kong industries performed relative to its competitors in terms of DFA. It was found that Japan consistently scored better design efficiencies than Hong Kong whereas Chinese products were only marginally inferior to Hong Kong products.

2.3 The UMass system can also be used to provide insight into where design improvements are best effected. This improvement process will of course require the application of general as well as product-specific heuristic knowledge (for instance, 62 common heuristics for assembly are discussed in [10].). In [9], it was demonstrated that it is possible to prioritise the heuristics in a given product family environment by logging the invocation of each heuristic during the design process and noting the percentage improvement achieved in the design efficiency. When the logging process is extended over a large number of products, it is possible to estimate the 'popularity' and 'power' of each heuristic.

3. GEOMETRIC RECOGNITION FOR DFA

3.1 An examination of the working charts of the UMass system reveals the need for two types of decision making. The first type requires the utilisation of expert knowledge concerning the product and the associated assembly processes. Hence, this type of decision making tends to be product-family specific. The assignment of insertion codes is dominated by decisions of this type. In the second type of decision making, the information needed is purely topological and geometric which, if the product was designed using a CAD system, should be available within the CAD data file. This CAD data however needs to be interpreted suitably to extract the specific dimensions, features, etc. needed for making each decision. For instance, the assignment of handling codes is dominated by decisions of this type. The principles underlying such geometric recognition are generic in the sense that they are applicable across a broad range of product families. The present paper addresses the problem of feature recognition required in decision making of the second type in the UMass system.

3.2 A wide range or approaches and algorithms for the geometric recognition of part features, mainly in the context of machining under numerical control, has been developed in recent years. The present paper describes a broad framework (see Figure 1) for the geometric recognition of the product features required in DFA from CAD data. The approach is essentially modular and eclectic in order to enable the appropriate utilisation of the wide range of algorithms developed by previous workers on the general problem of feature recognition. The following sections describe the modules in Figure 1, the associated contemporary algorithms and the proposed enhancements to suit the needs of DFA.

4. THE CAD MODEL

4.1 Current CAD systems support three types of modelling : wireframe, boundary, and solid. In wireframe modelling, the data base stores points, lines and circles. No information on the faces is available. In boundary representation (BR) the elements are planar polyhedra or curved surfaces. Faces are explicitly defined whereas features are implicit in the relationships between faces. Solid modelling uses solid primitives such as cylinders, cubes and tetrahedra. Some modern systems enable user-defined solid primitives to facilitate designing by features. However, such systems tend to be application-specific.

4.2 In assembly, one is most concerned with mating faces between pairs of parts. Further, an assembly related feature is recognised as a set of adjacent faces satisfying a specified set of topological and geometric inter-relationships. Hence, ready data on faces is particularly useful in DFA.

4.3 In the present work, AutoCAD Version 11, which provides a BR environment, is used to create the CAD model. The output from the CAD system is obtained as a DXF file (implementation for IGES format is in progress) and passed to subsequent modules for processing. Since IGES output is available in most CAD systems, it should be a trivial exercise to interface downstream modules to most CAD systems other than AUTOCAD.

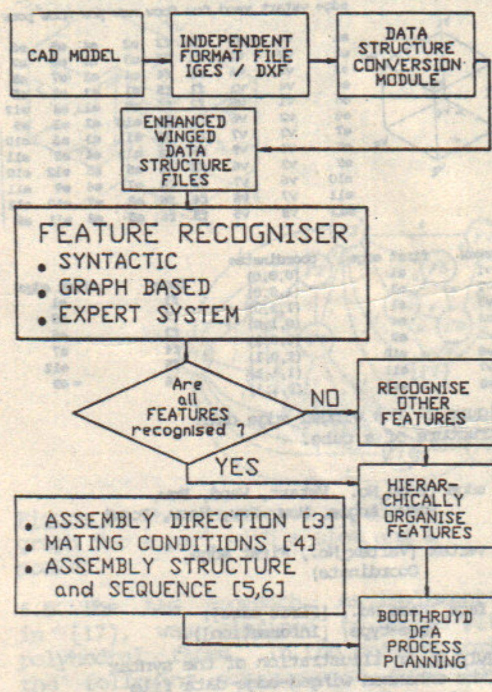
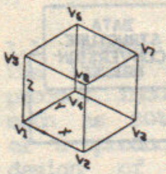


Figure 1 A broad framework for the computerisation of the geometric decision making needed in DESIGN FOR ASSEMBLY.

5. DATA STRUCTURE FOR DFA

5.1 The IGES/DXF file needs to be restructured to provide a format that is most convenient for feature recognition for DFA. The present work utilises an enhanced version of the Winged-Edge Data Structure of Baumgart [11].

The Winged-Edge Data Structure (see Figure 2) consists of lists of faces, edges and vertices. From each face, there is a pointer to each of its bounding edges. Likewise, there is a pointer from each vertex to each of the edges bounded by the vertex. Each edge is assigned a direction by specifying the starting and ending vertices (Vstart, and Vend respectively). Faces are classified as clockwise or anti-clockwise (fcw, or fccw respectively). The structure contains information on the next (ncw, and nccw) and previous (pcw, and pccw) faces. Thus, the relationships between adjacent faces, which are essential for feature recognition, are explicitly preserved.



edge	vstart	vend	fcw	fccw	ncw	pcw	nccw	pccw
e1	v1	v2	f1	f2	e2	e4	e5	e6
e2	v2	v3	f1	f3	e3	e1	e6	e7
e3	v3	v4	f1	f4	e4	e2	e7	e8
e4	v4	v1	f1	f5	e1	e3	e8	e5
e5	v1	v5	f2	f5	e9	e1	e4	e12
e6	v2	v6	f3	f2	e10	e2	e1	e9
e7	v3	v7	f4	f3	e11	e3	e2	e10
e8	v4	v8	f5	f4	e12	e4	e3	e11
e9	v5	v6	f2	f6	e6	e5	e12	e10
e10	v6	v7	f3	f6	e7	e6	e9	e11
e11	v7	v8	f4	f6	e8	e7	e10	e12
e12	v8	v5	f5	f6	e5	e8	e11	e9

vertex	first edge	coordinates
v1	e1	(0,0,0)
v2	e2	(1,0,0)
v3	e3	(1,1,0)
v4	e4	(0,1,0)
v5	e9	(0,0,1)
v6	e10	(1,0,1)
v7	e11	(1,1,1)
v8	e12	(0,1,1)

face	first edge
f1	e1
f2	e9
f3	e6
f4	e7
f5	e12
f6	-e9

Figure 2 The winged edge data structure of a cube.

```
edge (Edge_No., Vstart, Vend, Fcw,
      Fccw, Angle, Ncw, Pcw, Nccw, Pccw)

vertex (Vertex_No., First edge,
        Coordinate)

face (Face_No., [First edge],
      Face-type, [Information]).
```

Figure 3 An illustration of the syntax of the enhanced winged-edge data file in prolog unit clauses.

5.2 The following enhancements to the Baumgart Winged-Edge Data Structure have been made in the present work:

(i) Associate with each edge, the angle, θ, between the two faces adjacent to the edge - this facilitates the creation of Attributed Adjacency Graphs described later.

(ii) Include information on the 'first edge' for each topological entity such as lake, isle, etc. The enhanced structure thus has first edge lists instead of a single first edge entry.

(iii) Associate a surface-type-label (plane, cylindrical, etc.) with each face - this enables the handling of both polyhedral and curved faces.

(iv) Associate with each face, the parameters of the equation of the plane containing the face - this facilitates the determination of normal vectors needed for determining the direction of assembly.

The Enhanced Winged-Edge Data Structure is applicable when the faces are polyhedral, revolute or linearly extruded; the edges are straight or planar curves; and, whenever a plane face is adjacent to a linearly extruded surface, the plane face is orthogonal to the extrusion axis.

5.3 A compiler for converting the DXF files into the Enhanced Winged - Edge Data Structure (EWDS) has been written in PROLOG. The syntax of the EWDS file is illustrated in Figure 3. In the present version, the compiler only processes polyhedral faces. Implementation for curved faces is in progress.

6. FEATURE RECOGNITION

6.1 The following Table summarises the current approaches to feature recognition in general:

Recognition Approach	Current Application Domain
1. Syntactic Pattern Recognition [12]	3D depressions, protrusions, rotational parts, holes, pockets and slots.
2. State Transition Diagrams and Automata [13]	Steps, slots, pockets and rotational parts.
3. Volumetric Decomposition [14]	2 1/2 axis milling.
4. Expert Systems [15]	Holes, pockets, slots, planes, rotational parts, 2D section extrusions, 2D depressions, sheet metal work.
5. Set Theoric (CSG based) [16]	Cavity machining, features formed by union and subtraction of cylindrical parts.
6. Graph Based [17]	Polyhedral features, slots, pockets, etc. including disjoint, intersecting and nested features.

6.2 The present work is based on the Expert System Approach for recognising each feature from the geometric interrelationship between the faces consitituting the feature. Figure 4 illustrating this approach shows the PROLOG code for the recognition of a slot.

```
/* The rule to recognise a slot */
/* feature (slot, [End face 1, End
   face 2, Top face 1, Top face 2,
   Side face 1 of slot, Bottom face of
   slot, Side face 2 of slot])*/
feature (slot, [A,B,C,D,E,F,G]):-
adj (E, "plane", 0,F, "plane"),
/* A planar face E is adjacent with
   the planar face F, with adjacency
   attribute 0*/
adj (F, "plane", 0,G, "plane"),
adj (A, "plane", 1,E, "plane"),
adj (A, "plane", 1,F, "plane"),
adj (A, "plane", 1,G, "plane"),
adj (B, "plane", 1,E, "plane"),
adj (B, "plane", 1,F, "plane"),
adj (B, "plane", 1,G, "plane"),
adj (C, "plane", 1,E, "plane"),
adj (D, "plane", 1,G, "plane"),
A<>C,
B<>D.
```

Figure 4 Expert module for the recognition of a slot.

6.3 A Problem with the Expert System approach is that it uses a backward-chaining procedure where features are invoked one-by-one followed by an exhaustive search for the presence of the particular feature. Thus, computational time grows exponentially with the number of features invoked. This problem has been addressed in the graph-based study published in [17], where the idea of Attributed Adjacency Graph (AAG) was developed to enable a drastic reduction in the execution times for feature recognition.

6.4 An AAG of a part is a graph G = (N, A, T), where N is the set of nodes representating the N faces of a part, A is a set of arcs representing the edges contained in the part, and T is a set of attributes attached to the arcs. The attribute t_a of arc (edge) a, is set equal to 0 or 1, depending on whether the pair of faces intersecting at the edge form a concave or convex angle respectively.

Figure 5 shows an example of the AAG of a part with a slot and a pocket. The AAG is represented in the computer in the form of a matrix. Each node stores a pointer to the corresponding face information in the EWDS. The AAG facilitates feature recognition due to the fact that each instance of a feature results in the presence of a subset of the AAG for the part which satisfies a set of rules unique to the particular feature. Thus, in Figure 5, the AAG subset associated with the pocket has five nodes, has adjacency relationship as shown in the highlighted section of the Figure, and each arc of the subset has its attribute equal to 0.

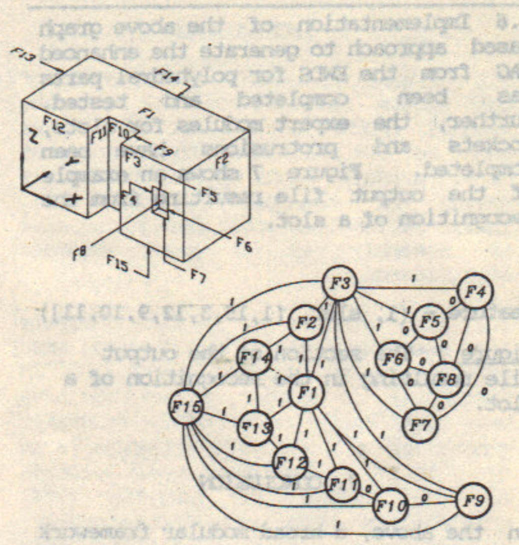


Figure 5 The attributed adjacency graph for a part with a slot and a pocket.

6.5 The AAG approach, as implemented in [17], was confined to parts with polyhedral faces. In the present work, the following enhancements have been made to overcome this limitation:

(i) Extend the attribute list (0,1) to (0,1,2,3,4) as illustrated in Figure 6 to describe different adjacency conditions (i.e. 2 : if $\Theta = 360°$; 3 : if $\Theta = 180°$; 4 : if $\Theta = 0°$)
(ii) Attach an attribute to each node to indicate the surface type to which the face belongs (e.g. plane, cylindrical, conical, etc.).

Figure 6 illustrates the AAG resulting from the above enhancements for a part with polyhedral and curved faces.

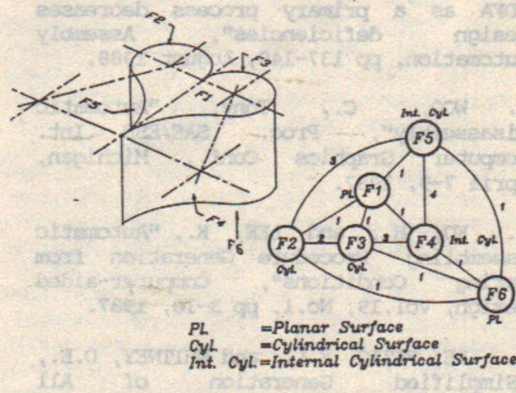


Figure 6 The enhanced attributed adjancency graph for a part with curved faces.

6.6 Implementation of the above graph based approach to generate the enhanced AAG from the EWDS for polyhedral parts has been completed and tested. Further, the expert modules for slots, pockets and protrusions have been completed. Figure 7 shows an example of the output file resulting from the recognition of a slot.

```
feature - (1, slot, [1,15,3,12,9,10,11])
```

Figure 7 The section of the output file resulting in the recognition of a slot.

7. CONCLUSION

In the above, a broad modular framework for the geometric recognition of the features encountered in DFA is presented. The modular approach, based on the concepts of expert systems and the Attributed Adjacency Graph, enables the progressive expansion of the feature library. The software being developed utilises a number of feature recognition algorithms proposed by previous workers in other areas with suitable enhancements to suit the needs of DFA. These enhancements have been briefly discussed.

REFERENCES

1. A.G. Andreev, et al, "Devising a Range of Types of Unified Sub-Assemblies for the Modular Design of Assembly Equipment," Vestnik Mashinostroenia, Vol.66, Issue 6, pp 32-36, 1986.

2. P.J. Sackett and A.E.K. Holbrook, "DFA as a primary process decreases design deficiencies", Assembly Automation, pp 137-140, August 1988.

3. WOO, C., Tony, "Automatic Disassembly", Proc. SAE/ESD Int. Computer Graphics Conf., Michigan, April 7-9, 1987.

4. KO, H., and LEE, K., "Automatic Assembling Procedure Generation from Mating Conditions", Computer-aided Design, Vol.19, No.1, pp 3-10, 1987.

5. DE FAZIO, F.L., and WHITNEY, D.E., "Simplified Generation of All Mechanical Assembly Sequences", IEE J. of Robotics and Automation, Vol. AA-3, No.6, pp 640-658, 1987.

6. GAIROLA, A., "Design Analysis for Automatic Assembly", Int. J. Prod. Res., 1986 Vol.24, No.4, 839-849.

7. KROLL, E., LENZ, E., WOLBERG, J.R., "A Knowledge-based Solution to the Design-for-Assembly Problem", Manufacturing Review, Vol.1, No.2, June

8. BOOTHROYD, G., and DEWHURST, P., Product Design for Assembly, Dept. of Ind. & Mfg. Engg., University of Rhode Island, Kingston, 1989.

9. P.K. Venuvinod, "Computer Aided Approaches to Product Design for Ease of Assembly," Autotech '87 Int. Conf. Industrial Automation, Hong Kong, October 1987.

10. M.M. Andreasen, et al, Design for Assembly, IFS, 1983.

11. Martti Mantyla, "An Introduction to Solid Modelling", Computer Science Press, pp101-112, 1988.

12. STALEY, S.M., et al, "Using Syntactic Pattern Recognition to Extract Feature Information from a Solid Geometric Data Base", Computers in Mech. Eng., pp61-66, 1983.

13. Milacic, V.R., "SAPT - expert system for manufacturing planning, in Computer Aided/Intelligent Process Planning, ASME Winter Annual Meeting, PED Vol.19, 1985.

14. Grayer, A.R., "The Automatic Production of Machined Component Starting from a Stored Geometric Description", Advances in Computer Aided Manufacturing, North Holland Publishing Co., Amsterdam, pp 137-151, 1977.

15. Henderson, M.R., "Automated Group Technology Coding from a Three Dimensional CAD Database", Symposium on Knowledge Based Expert Systems for Manufacturing, The Winter Annual Meeting of ASME, Anaheim, Ca., pp 195-204, 1986.

16. Lee, Y.C., and Fu, K.S., "Machine Understanding of CSG : Extraction and Unification of Manufacturing Features: IEEE Computer Graphics and Application, Vol.7, No.,1, pp 20-32, 1987.

17. JOSHI, S., and Chang, T.C., "Graph Based Heuristics for Recognition of Machined Features from a 3 D solid model", Computer-aided Design, Vol. 20, pp 58-66, 1988.