Technical Note 3

MMCs and PPCs as constructs of curvature regions for form feature determination

Ratnakar Sonthi1* and Rajit Gadh2

Shape feature information about a part is required to analyze the part for downstream issues such as manufacturability and assemblability. One method of obtaining the feature information is feature recognition from the geometric model. The current approach called Curvature Region (CR) approach for feature determination from solid models categorizes features into two primitive shape classes: protrusions and depressions. In the first step, these primitive shape classes are recognized from the solid model. In the next step, the primitive shape classes are matched with feature definitions to obtain features. © 1999 Published by Elsevier Science Ltd. All rights reserved

DESCRIPTION OF THE ALGORITHM

The information present in the CAD model representation of a geometric part is very detailed. As a result, obtaining geometric feature information directly from the CAD model is a cumbersome process. Hence, in our approach, the b-Rep is mapped to an alternate and simpler representation called Curvature Region Representation (CR-Rep) which has information essential for obtaining the primitive shapes.

The CR-Rep is represented by a graph of curvature regions where a curvature region is defined as a set of connected points all of which have the same signs for their principal normal curvatures1. The data structure used for a curvature region has information about the type of the curvature region and the parent B-Rep entities of the curvature region. As shown in Figure 1, the nodes in the CR-Graph have pointers to their corresponding B-Rep elements (faces, edges and vertices). For the object shown in Figure 1 the depression comprises of the faces F1 and F2, edge E and vertices V1 and V2.

Each face and edge is mapped one or more curvature regions and a vertex is mapped to a single curvature region. A face is mapped to its curvature region(s) by considering the principal normal curvatures of the points on the face. An edge is mapped to its curvature region(s) by considering the concavity/convexity and the tangential curvature of the points and the edge. The concavity/convexity of the edges incident at a vertex are used to determine curvature region of a vertex.

The connectivity of the curvature regions is represented by the edges of the graph (called CR-Graph) representation of the CR-Rep. The CR-Graph is subsequently operated upon by a set of rules to obtain primitive shapes and features. After the application of the rules a primitive shape or a manufacturing feature comprises a sub-graph of the CR-Graph. Since each curvature region contains pointers to its parent B-Rep entities, the B-Rep entities corresponding to a recognized feature can be readily obtained by following the pointers in the CRs. The algorithm for the approach is as follows:

Start main program
findCRGraph(in B-Rep, out CR-Graph); // This determines the CR-Graph for the part
simplify CRGraph(in CR-Graph, out simplifiedCR-Graph); // Simplifies the CR-Graph
determinePrimitives(in simplifiedCR-Graph, out Protrusions, out Depressions);
defineFeatures(out definedFeaturesArray); // Features defined as graphs
determineFeatures(in Protrusions, in Depressions, in definedFeaturesArray, out foundFeaturesArray); //Determine features from protrusions/depressions
determineFeatureDimensions(in-out foundFeaturesArray);
End

*Corresponding author
1 I-CARVE Lab, Department of Mechanical Engineering, University of Wisconsin—Madison, Madison, WI 53706, USA
2 I-CARVE Lab, Department of Mechanical Engineering, University of Wisconsin—Madison, Madison, WI 53706, USA
Paper Received: 20 April 1998. Revised: 4 August 1998. Accepted: 4 August 1998
The findCRGraph procedure is as follows:

Start

findCRGraph(in B-Rep, out CR-Graph)

createInstanceofCRGraph(out CR-Graph);

foreach entity E = Face(F) or Edge(E) or Vertex(V) in B-Rep

DetermineCurvatureRegion(in E, out CR); //Find the curvature region of E

AddNeighboursToCR(inout CR, in neighborArray); //Add neighbors

AddCRtoCRGraph(in CR, in-out CRGraph); //Add the CR to CR-Graph

End //End of loop

End

The current implementation is on ProEngineer/Pro-Develop and the software extracts protrusions and depressions and seven basic features (rib, wall, tab, boss, pocket, blind hole and through hole) from a model. Protrusions and depressions are represented as CR-Graphs in the computer. A sub-graph isomorphism is performed between the CR-graphs of the protrusions and the CR-Graphs of rib, tab, wall and boss features. Similarly, a sub-graph isomorphism is performed between the CR-Graphs of the depressions and the CR-Graphs of pocket, blind hole and through hole. A feature is recognized when a sub-graph isomorphism is found.

RESULTS

For the following four parts the features obtained in the current implementation are indicated. The features obtained from depressions are highlighted in red and the features obtained from protrusions are highlighted in blue. The reader is referred to Sont2 for the features obtained on the other benchmark parts.
UK PART1 AND UK PART2

Using the CR-Approach, the list of feature classes found on the object in Figure 2 are: four through holes, seven blind holes and 6 through slot. A through slot is defined to comprise three flat faces joined by two concave edges or concave fillets. The two through slots obtained in the bottom left figure are evident from the definition. However, for the figure on the bottom right, four through slot features are obtained as indicated, instead of a single pocket feature. Moreover, a few features are not determined due to feature interaction. Similarly, for the object in Figure 3 the list of features obtained are: one through slot (feature highlighted in red on the left), four through holes and four blind holes.

KIM PART1 AND KIM PART2

For the object in Figure 4, the list of features obtained are: one boss, 9 through holes and 8 blind holes. Some features on the part are not determined currently due to the presence of feature interactions. New definitions for variations of rib and slot features are required to determine slot and rib features on the part.

For the object illustrated in Figure 5, the list of features obtained are: 5 ribs, one boss, 3 slots and a through hole.

REGLI PART

The strength of the CR-Approach lies in handling non-linear surfaces or filleted surfaces in a model. For example, for the object shown in Figure 6, there are several filleted surfaces and a non-linear surface. A partial list of primitive feature classes obtained for the object comprises: Depressions A, B, C and D and Protrusion A. Depression A has a non-linear face which is entirely concave. The non-linear face in the original model is split into two regions: convex-flat and concave-flat. Depression A is analyzed to obtain a generic pocket feature. Depressions of type B result in blind holes. Depression C results in a slot feature. Depression D results in a rectangular pocket. Protrusions of type A result in rib features. Therefore the features obtained for this part are: 2 ribs, 1 rectangular pocket, 1 generic pocket, 1 through slot and 4 blind holes.

SUMMARY

This technical brief presents the results of an approach to...
obtain features from solid models using the notion of Curvature Regions 1,2. The result of the application of the approach on nine sample parts presented at CIE97 is described in this article. The features found using the current approach are mainly applicable for near net-shape manufacturing processes such as injection moulding and die-casting. Furthermore, the current approach does not determine interacting features from parts hence some features on the nine parts are not determined. The determination of interacting features is currently being researched.

ACKNOWLEDGEMENTS

This work was funded by NSF research Initiation Grant No. DMI 9410359.
REFERENCES


Ratnakar Sonthi received his BS in Mechanical Engineering from Indian Institute of Technology, Madras. He is currently a research assistant working towards his Doctoral Degree in Mechanical Engineering at the University of Wisconsin—Madison. His research interests include Feature Recognition, Internet-based CAD/CAM and Computer Graphics.

Rajit Gadh is an Associate Professor at University of Wisconsin—Madison. His research interest is Virtual Design/Prototyping. He received his BS from Indian Institute of Technology, Kanpur, MS from Cornell University and PhD from Carnegie Mellon University. He has been a visiting researcher at University of California, Berkeley and has spent roughly five years in industry working for companies such as Ford, GM and DEC. Professor Rajit Gadh’s research interests are in the modeling of fundamentally new Computer-Aided Design approaches. The first theme, virtual shape design involves development of a fundamentally new interface and geometric representation to support designing shapes within a virtual reality system. The second theme, shape abstraction definition and extraction to support design, involves determination of shape features from component shapes and assemblies in the context of Computer-aided Design. He has received the 1996 AT&T–Lucent, Industrial Ecology Fellowship, the NSF CAREER Development Award and Grant in 1995, in 1994 the Engineering Foundation, Research Initiation Award, and the Ralph Teetor Research and Educational Award, from SAE in 1993.