

FEATURE-BASED NON-MANIFOLD MODELLING SYSTEM FOR INTEGRATION OF CAD AND CAE SYSTEMS FOR INJECTION MOULDING PRODUCTS

Sang Hun Lee
Graduate School of Automotive Engineering
Kookmin University
Seoul, Korea
shlee@kookmin.ac.kr

ABSTRACT

A feature-based non-manifold geometric modelling system has been developed to provide an integrated environment for design and analysis of injection moulding products. In this system, the geometric models for CAD and CAE systems are represented by a non-manifold boundary representation and are merged into a single geometric model, from which the suitable form of geometric model for design and analysis can be extracted. In our system, a part model containing geometric and feature data is first created using the feature modelling shell. Then, the abstract model for analysis is extracted using the feature mapping shell. Finally, two or three node meshes for injection moulding simulation systems are automatically generated on the abstract model. By introducing the Boolean operations based on a non-manifold representation, the feature deletion and interaction problem of the feature-based design system has clearly been solved. The sheet modelling capabilities were also developed for easy modelling of thin plastic parts.

KEYWORDS : injection moulding, non-manifold geometric modelling, feature-based design, feature mapping.

INTRODUCTION

(1) Background and Objective

Traditional design and manufacturing process for plastic injection moulding parts is performed based on the expert's several years of experience, and its long-term trial-and-error causes an increase in costs. In order to reduce costs, there have been various trials to develop a specialized CAD system for plastic part and mould design [1, 2, 3], and to develop a CAE system to simulate the injection moulding process to find defects before the manufacturing stage [4, 5, 6, 7]. Traditional design and manufacturing processes can be supported by

the CAD and CAE systems as shown in Fig. 1. At the initial design stage, the specialized CAD system can help designers. The process simulation with the CAE system verifies the initial design result. The mechanical capabilities and defects of a plastic part are predicted through the simulation process. If the simulation results do not satisfy the functional requirements, the design process is repeated by feeding back the simulation results. The re-design process is repeated until the functional requirements are satisfied.

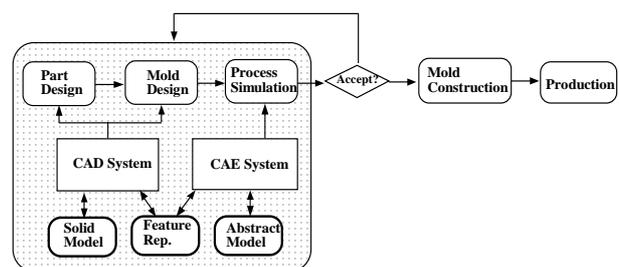


Fig. 1 Design process using traditional CAD and CAE systems for plastic injection moulding.

The geometric models necessary for the initial design stage are solid models containing feature representations such as ribs and bosses, whereas the models needed for the simulation process are shell meshes made from a sheet model composed of medial surfaces and wireframes that are contractions of the solid. Since the current CAD systems only deal with solid models, and the abstract model is generated in the CAE system using a pre-processor, the designer must create two types of models at the same time to design and verify a plastic product. Therefore, if two geometric models for design and simulation are created in one CAD system at the same time, the CAD and CAE systems can be integrated, and concurrent engineering methodology can be realized more readily.

We developed a feature-based CAD system based on a non-manifold geometric modeller as shown in Fig. 2 to produce the two models. In this system, the

geometric models for CAD and CAE systems are represented with a unified data structure and stored in a part model. The solid model for design, or the abstract model for analysis, can be extracted from this part model directly. Since the part model also contains the removed faces, during feature-based modelling, the feature deletion and the feature interaction checking were easily implemented.

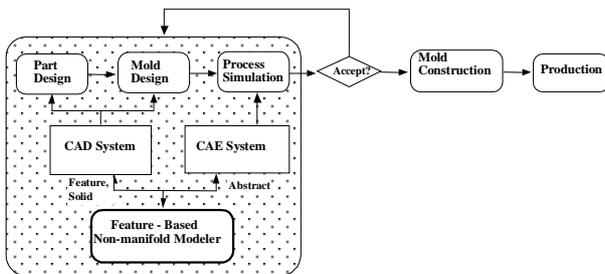


Fig. 2 Integration of CAD and CAE systems by introducing unified geometric modeling.

(2) Related Work

Huh and Kim [1] developed a knowledge-based CAD system to support the initial design of injection moulding products. This system contains two components: the one is an expert system for the optimal design of ribs and gates, the so-called RIBBER and GATEWAY, and the other is a three-dimensional geometric modelling system to represent design result in solid models. In order to develop the expert system, they gathered empirical equations and knowledge for design of ribs and bosses, and extracted the rules for the knowledge-based modules. Ishii et al. [2] proposed a system based on the design compatibility analysis methodology, in which product design is examined to determine how well it meets customers' requirements and the constraints on mould production and the injection moulding process. Gadh et al. [8] emphasized the role of the systems based on experts' knowledge to judge the mouldability for products as an alternative to numerical analysis systems. They also mention the representation and extraction of features for knowledge-based expert systems.

The injection moulding process is composed of a series of filling, packing and cooling processes. There has been considerable research to predict defects and optimal injection moulding conditions [4, 5, 6]. As a result, there have appeared various commercial CAE systems: MOLDFLOW developed by Austin et al. [9], and C-FLOW developed by Wang et al. through the Cornell Injection Moulding Program [4, 7]. These systems focus on the filling simulation, but recently packing and cooling process simulation modules have been included. MOLDFLOW includes not only

MF/FLOW but also MF/COOL and MF/WARP.

DESIGN OF AN INTEGRATED CAD/CAE SYSTEM FOR INJECTION MOULDING

(1) Functional Requirement

As described above, an integrated CAD/CAE system for the design of plastic injection moulding parts should have the following functionality in order to support feature and solid models as well as abstract models for CAE systems for injection moulding process simulation and structural analysis.

- Create thin-walled parts easily. The objects dealt with in this paper are thin-walled plastic parts, especially outer panels for electronics devices. They have large surfaces as well as thin and constant thickness walls. Therefore, powerful modelling capabilities for thin-walled parts should be provided in the system [10].
- Provide design-with-feature capabilities [11]. When a plastic part is designed, first the main shape is determined and then the size and location of sub-features, such as ribs and bosses, to satisfy functional requirements are determined [1]. Thus, the three-dimensional CAD system for plastic parts must allow initial design and re-design with features, and represent the design result with a solid model.
- Provide geometric models for analysis efficiently. While a part is designed, the injection moulding process simulation is performed to find the problems in advance, and the result is fed back to modify the initial design. Shell meshes on the medial surface of a part should be derived easily in order to run the simulation packages. In order to generate shell meshes, at first, the main shape of a part is converted to a sheet model of medial surface, and then sub-features are converted to sheets or wireframes depending on mesh size. The geometric model composed of sheets and wireframes, on which meshes are generated, is called an abstract model. An example of an abstract model is shown in Fig. 3.

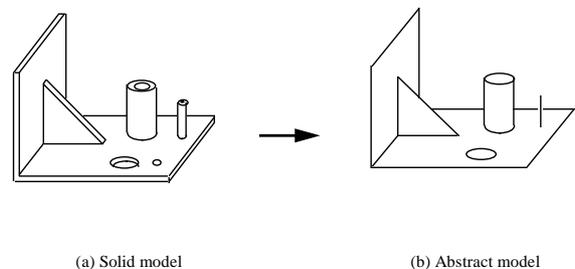


Fig. 3 Example of abstract model for analysis of injection moulding.

(2) Possible CAD/CAE Integration Methods

When we construct an integrated CAD/CAE system for an injection moulding product, satisfying functional requirements, based on current research achievements the possible combinations of integration methods can be classified into three groups according to model types to be stored in the database of the system for the design result:

- Store only the solid model
- Store solid model and feature information
- Store solid model, feature information, and abstract model for analysis simultaneously

If one wants to obtain an abstract model for analysis from the design database, a conversion or extraction process is required, according to storage method, as shown in Fig. 4. Now we investigate the abstract model generation process for each method, and compare each of these to show the advantages and disadvantages of CAD/CAE integration.

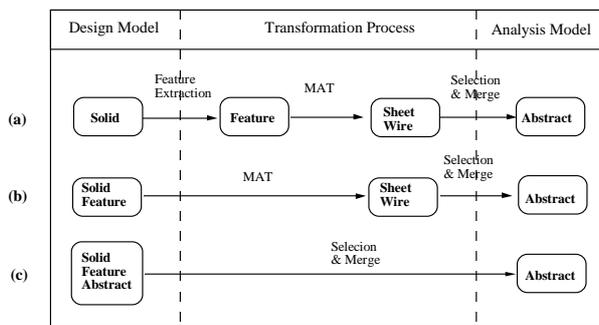


Fig. 4 Extraction of the abstract model for analysis from the design model.

In the first method, which stores only the solid model, an abstract model for analysis must be generated from the solid model to obtain shell meshes for the CAE system. In order to get an abstract model, sub-features such as ribs, bosses, or holes must be extracted from the solid model and transformed into sheets or wires according to the given mesh size, and then the medial surface of the main shape is generated through medial axis transformation (MAT) [12]. The abstract model is completed by combining sheets or wires for sub-features and a sheet of medial surface for the main feature. This process requires considerable computing time and it must be repeated for the whole body whenever any partial modification is performed, if the design is performed in parallel with the analysis in a concurrent engineering environment. Moreover, since the design result does not contain feature information in this method, the designer must change the shape of the solid model using the solid modelling system directly when modification is necessary.

In the second method, the design for plastic parts is performed on a feature-based modelling system and the solid model and feature information for the design are stored in the database. When the abstract model is required for analysis, the medial axis transformation is applied to the sub and main features, respectively, to produce a sheet or a wire according to the mesh size. Then the final abstract model for analysis is obtained by combining sheets and wires extracted from the features. Comparing with the first method, storing only solid models, this method utilizes feature information in the database so that feature extraction can be omitted. However, this method still has a time-consuming process when MAT is performed for each sub-feature whenever an abstract model is required during the design process.

In the third method, the database stores not only solid models and feature information, but also the abstract models for the features at the design stage. As shown in Fig. 4(c), the solid model and abstract model for each feature are generated during the feature-based design, and the analysis model can be extracted directly from the database according to the mesh size. In this method, a design change is propagated to the analysis model instantly so that an abstract model can be extracted directly for analysis.

(3) Design for Integrated CAD/CAE System

Of the three CAD/CAE integration methods, the third method alone satisfies the requirements for a concurrent engineering environment. In order to build an integrated CAD/CAE system based on the third method, we determined the system specification as follows:

- Provide the feature-based modelling capabilities and the feature conversion capabilities from design features to analysis features, which are the abstract model for shell mesh generation. As illustrated in the system functional requirement, the CAD system for injection moulding products should be able to support feature-based design. Moreover, since the analysis features for CAE systems can be easily obtained from the design feature information, feature mapping from design to analysis features is more efficient than feature extraction of the analysis features from a solid model.
- Use non-manifold representation for the geometric modelling system. The non-manifold B-rep can represent not only solid objects but also wireframe, sheet, and cellular objects, along with a mixture of these, in a single data structure [13, 14, 15]. Thus, solid models, for design, as well as abstract models, for analysis, can be represented in a single modelling environment. Moreover, when the user models thin-walled plastic parts, they can create a

sheet model for the inner or the outer wall, and then offset it to generate a thin-walled solid with a given constant thickness. In addition, since a non-manifold B-rep supports cellular models in which the hidden faces of features deleted with the Boolean operations can be stored, any feature can be cancelled immediately independent of the generation sequence, and feature interaction can be easily detected [15, 16].

- Use the part model in which the solid and abstract models for all features are merged into a non-manifold model. Not only a solid model but also an abstract model for each feature is inserted into the part model so all the geometric information for design and analysis can be contained in a single part model. By adopting this method, design change propagates to the analysis model immediately and the abstract model, for analysis, can be obtained by simple filtering from the geometric model for the part if analysis is necessary during design.
- Use the object-oriented programming technique. Since a feature-based modelling system uses redefined features, it is difficult to enlarge the feature library when the user needs to define a new feature. To solve this problem, we adopted the currently widely used object-oriented programming technique so that the user can add a new feature to the feature library without changing the existing system code.

The integrated CAD/CAE system that was implemented according to the above design strategy consists of the following five sub-systems as shown in Fig. 5:

- User interface
- Feature modelling shell
- Feature mapping shell
- Non-manifold geometric modelling system
- Product database

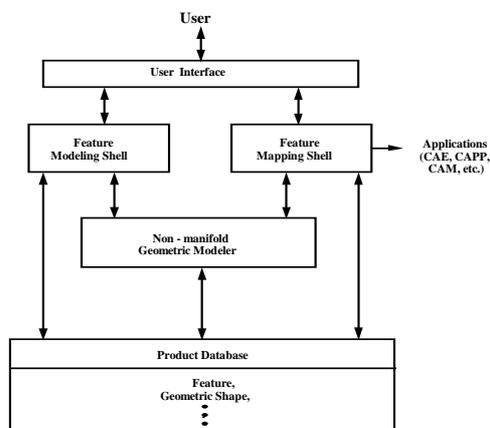


Fig. 5 Architecture of the integrated CAD/CAE system.

Of these sub-systems, we briefly explain three main sub-systems: the feature modelling shell, the feature mapping shell, and the non-manifold geometric modelling system.

The feature modelling shell creates, deletes and modifies features and maintains the hierarchical relationship among features. This module sends messages to create or delete geometric models for features to the non-manifold geometric modeller.

The feature mapping shell converts features for design to features for other application areas, such as CAE, CAM, CAPP, etc. In this paper, the only commercial package for mould flow simulation, C-FLOW, is developed. Development of feature mapping modules for other applications remains for future work.

A non-manifold geometric modelling system, called AnySHAPE [14], receives messages of creation, deletion, query from the feature modelling or mapping shells, and performs corresponding modelling operations. Especially, this modeller manages the merged-set models for parts, which are generated by merging solid models and abstract models for all features. In this system, the capabilities for sheet modelling and transformation into solids are provided for simple modelling of thin-walled parts.

The design and analysis process with this system is as follows. First, the user creates the solid model for the main shape of the part. They can use not only traditional solid modelling functions but also the sheet modelling capabilities to accelerate thin-walled part modelling. After generation of a solid model, the user prepares a sheet model for the medial surface of the part through an automated MAT procedure or manual work. They can use a sheet model before transformation into a solid. Secondly, the user transfers to the feature modelling shell and registers the solid and sheet models for the main shape of the part as a base feature, and then creates sub-features sequentially to complete the whole part. Finally, the user goes to the feature mapping shell. In this module, the abstract model for analysis is extracted and two or three node meshes are automatically generated on this abstract model. These meshes are used as a geometric input for the C-FLOW.

FEATURE MODELLING SYSTEM

(1) Representation of Features

The part model contains data for the features and geometric information. Features are connected to each other with a graph structure in which features have a parent-child relationship. The base feature for the main shape of the part is the starting node of the graph.

The creation sequence of features is recorded in the attribute area of the part model. The geometric model is represented with a merged-set model that is the result of a sequence of the Boolean operations among the features. The merged-set model is a non-manifold model that stores all topological entities of primitives for features, and entities generated by intersection during the Boolean operations on implementation of features. The topological entities in the merged-set model all have records of their own birth. These records are used for feature deletion, feature intersection checking, abstract model extraction, etc.

The data for features consist of common data for all features and specific data for the different features. That is, a feature is defined as two levels of classes, i.e., a base class, Feature, and its derived class, Rib or Boss, for specific data for different features. The base class, Feature, stores common data for all features and the derived class such as Boss or Rib defines specific data for each feature and inherits common data from the base class. The Rib class contains the variables for the cross-section of a rib, that is, thickness, height, draft, and fillet radius. The Boss class includes variables for height, wall thickness, bottom thickness, and draft.

In addition, the common method for all features such as `create_solid_model()` and `get_mapping_model_for_cflow()`, are declared as virtual functions in the Feature class and implemented at the derived classes similarly to Boss and Rib. If the system sends a message to execute these functions to Feature object, considering the type of feature contained in the feature instance, the function defined in the derived class is executed. Thus, the user can add a new feature to the feature library without a change of system code because they only implement the common method for the new feature and compile and link it to the existing system.

Note that a feature can include two, or more, abstract models whose shapes can be wireframe, sheet, or solid. For instance, a boss can be abstracted into a wireframe or a cylindrical sheet. The abstract model for a depression feature such as a hole is just a solid. The type of abstract model depends on mesh size. In this system, all possible abstract models are stored in the part model, and one of them satisfying the condition is selected and extracted.

An example for the part model is illustrated in Fig. 6. The part in Fig. 6(a) has a box type of base feature and two sub-features, rib and boss. The geometric model for the part is a non-manifold model merging solid models as well as sheets and wireframes as abstract models. The representation for the part in this system is shown in Fig. 6(b). The part instance, **part1**, stores the pointer to base feature, **base1**, and to geometric model, **part1_model**.

Features are connected each other with a graph where the parent-child relationship is defined. Each feature points to the solid and abstract models.

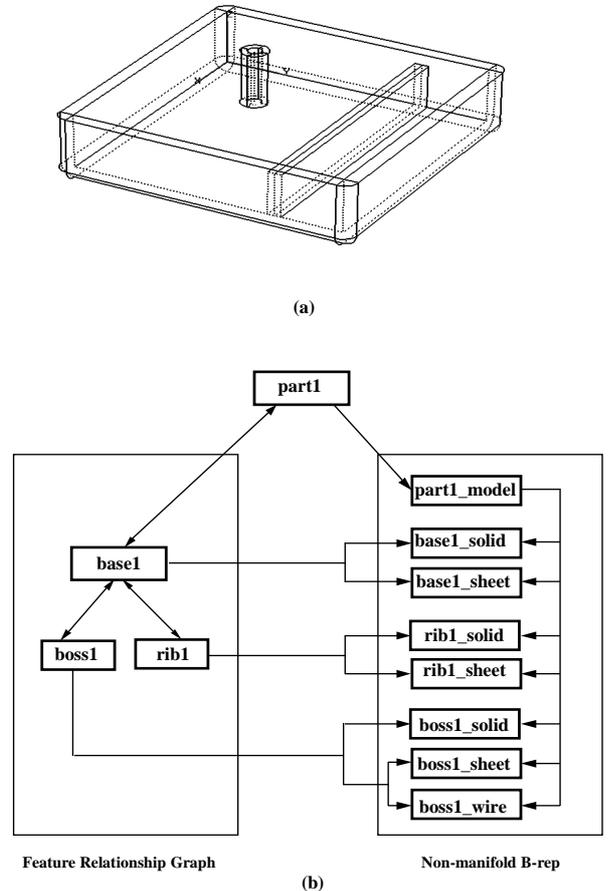


Fig. 6 Example of part representation

(2) Feature Creation

In the part modelling process the base feature for the main shape of the part is created first and then adequate sub-features are implanted sequentially. However, the user cannot use geometric modelling capabilities such as Boolean operations in the feature modelling shell and they can only add or delete features. Now we explain the part modelling process in detail.

First, the user creates a solid model and an abstract model for the main shape of the part using the geometric modelling capabilities in the non-manifold geometric modelling system. Since the main shape is a thin-walled solid object, the user can create a sheet model for the outer or the inner wall using the sheet modelling capabilities in AnySHAPE [14] and then apply the automatic transformation function from a sheet to a solid with the given thickness. The abstract model for the base feature can be obtained through the medial

axis transformation, but this system does not provide the MAT function. If offsetting the sheet model of the outer or the inner wall generates the main shape, this sheet can be used as the abstract model for the base feature with some modification. Simultaneously with the registration of the main shape, a new part model is generated, and performing the Boolean union operation between the solid and the abstract models in the system creates the merged-set model.

Finally, the user creates the sub-features at adequate locations. In order to create sub-features, the user selects the feature from the menu and inputs data requested by the system. Then the system creates an instance for the new feature class and fills the record with input data, and connects features in the graph. Then the system creates the solid and abstract models according to the shape variables, and merges these two models with the merged-set of the part model using the Boolean operation. At this point, the abstract model for the sub-feature is extended to reach the sheet model for the base feature.

(3) Feature Deletion

When the user orders the deletion of a feature, the system removes all vertices, edges, faces and regions originating from the solid and abstract model of the specified feature, and then eliminates the feature from the feature graph considering the hierarchical relationship with other features. In this system, if a parent feature is deleted, the child features are also deleted. If a feature has multiple parents, it would be deleted when all the parents are deleted. After a feature is deleted, the entities suppressed by that feature are revealed in the part geometry. It is simple to recover the removed entities because the non-manifold merged-set model stores all the topological entities and their historical record created during design.

FEATURE MAPPING

In the feature mapping shell, the system converts design features to features for application areas such as CAE, CAPP, and NC processing. In this paper, we employed the methodology of Rogen et al. [17] for feature conversion and developed a feature mapping shell to convert primary design features to the secondary features for application. In this paper, we develop the mapping module for C-FLOW, which is one of the commercial CAE packages for simulation of the injection moulding process. The development of the mapping modules for the other applications will be the subject for future work. Now, we review the model

necessary for C-FLOW, that is, the secondary features, and then explain how this system generates the secondary features for C-FLOW.

The method to make the analysis model for C-FLOW from the solid model for the plastic part is introduced in detail in the C-FLOW Users' Guide [7]. According to the manual, C-FLOW receives six types of FEM meshes as geometric input: linear triangle, cold/hot solid runners, cold/hot annular runners, and connector. The linear triangle is a three-node shell mesh, and the others are all two-node beam meshes. Three node meshes are used to represent the main shape and ribs, and two node meshes are used to represent bosses, pins, runners, and gates. These meshes are understood as the secondary features for C-FLOW. Thus the feature mapping system performs operations to convert the design features such as rib, boss, hole, etc., to the secondary features for C-FLOW such as two- or three-node meshes. The conversion process is composed of four main steps as follows:

- (Step 1) Extracting the necessary information to generate the secondary features. As mentioned in the C-FLOW Users' Guide, the design information to be considered for mesh generation is coincident with the design features such as rib, boss, hole, runner, and gate, which are used during the design stage. Since design is performed in the feature-based modelling system, any feature extraction process is not necessary and the design feature information created during the design stage can be used directly.
- (Step 2) Selecting the features and topological entities of the merged-set model for the part that contribute to creating the abstract model. Selecting features follows the method instructed by the C-FLOW Users' Guide, and selecting topological entities from the part model follows the selection algorithm for the non-manifold Boolean operations.
- (Step 3) Generating an abstract model for analysis that is composed only of the topological entities selected in the previous step.
- (Step 4) Generating shell meshes on the abstract model produced in the previous step.

Now we explain the second and third steps in more detail.

(1) Selecting Topological Entities for the Abstract Model

Here the system selects and marks the topological entities in the part model that contribute to building the abstract model for analysis. First, the system selects one abstract model participating in the construction of the abstract model of the part, from the abstract models

of each feature. Because some features can have multiple abstract models, the system selects one of them, or does not select any to exclude the feature. For example, the boss feature has a wireframe and a sheet as abstract models and they are included in the part model at the design stage. If the mesh size selected is small, the wireframe is selected. Otherwise, the sheet is selected as an abstract model for the feature. In the case of holes, the hole is omitted if the hole size is smaller than the given mesh size. In the program, the member function of the feature for selecting an abstract model is implemented to return the pointer to the adequate abstract model with respect to the given mesh size.

Secondly, the system generates the CSG tree of the abstract models of selected features considering the order of creation of the features stored in the part model. The entities to be included in the analysis model are determined with this tree among the topological entities of the merged-set model of the part. In order to build the CSG tree, a string is generated, which consists of the names and signs of the abstract models of the features considering the order of feature creation and the types of features. If the type is a protrusion a '+' sign is assigned, and if a depression a '-' sign is assigned. Then the system interprets the string and generates the CSG tree. Finally, the system selects and marks the entities to be alive for construction of the analysis model from the topological entities of the merged-set model of the part according to the CSG tree of abstract models.

For instance, the CSG tree for the simple part shown in Fig. 6 is (base_sheet + rib_sheet) + boss_wire when considering the feature type and creation order. The selected entities by the CSG tree are shown in Fig. 7. The selection process is explained by Masuda [15]. Note that the image on the screen appears as if the analysis model is complete, but the unselected entities remain in this model even though they are not displayed.

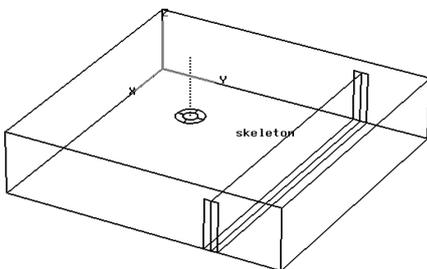


Fig. 7 Selection of topological entities to build the abstract model.

(2) Generation of Abstract Model and Meshes for Analysis

After the selection is finished, each vertex, edge, face and region of the merged-set model is marked as alive or dead. To generate FEM meshes, a new model that does not contain the unnecessary topological entities marked 'dead' should be generated. One can build this model by creating entities coincident with the entities marked 'alive' with the Euler operators. However, it is easier than the above to copy the merged-set model and delete the entities marked 'dead' with the Euler operators. We adopted the second method.

When the entities marked 'dead' are deleted, first the faces and then the edges and finally vertices are marked. The Euler operators to delete faces are KFMC and KFR, for edges, KEV, KEC, KEMS, and for vertices, KVS [14]. Next, the system deletes the redundant topological entities to clean the model. To clean the model, two adjacent faces with the same planar surface are merged into one, and two adjacent edges with the same straight line are merged into one. The cleaned abstract model for the marked model in Fig. 7 is shown in Fig. 8. The triangular shell meshes are generated once the abstract model for analysis is complete. The C-FLOW prefers uniform and regular triangular meshes. The nodes on the boundary edges of a face must be coincident with those of the adjacent face. As the algorithms for shell mesh generation are already published, we simply select one of them.

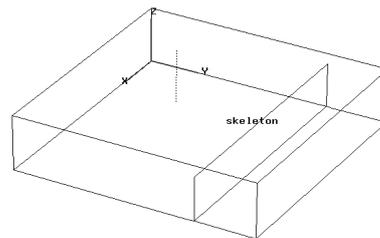


Fig. 8 Extraction of the abstract model from the merged-set of part.

CONCLUSIONS

Current CAE systems for simulation/analysis of the injection moulding process require a geometric input, the abstract model, which is composed of sheets for the medial surface and wireframes for bosses. However, the design result generated on the CAD system is a solid

model. Thus the user must create the abstract model again as they cannot use the solid model. To solve this problem and offer an integrated environment for design and analysis of plastic injection moulding products, we developed the feature-based design system based on a non-manifold geometric modeller.

In this system, the geometric models for CAD and CAE systems are represented with a single non-manifold boundary representation. The database stores not only all solid and abstract models for features, but also all topological entities and their history records generated by the Boolean operations so the system can provide the user with the solid model for design and the abstract model for analysis, immediately, if necessary. In addition, using this information in the database, we easily implemented immediate feature deletion and the detection of feature interaction, which are difficult to implement with a solid modeller. Moreover, this system is implemented using the object-oriented programming technique, so the user can define and add a new feature into the feature library without changing the source code of the existing system. In addition, the non-manifold geometric modeller provides the sheet modelling capabilities and transformation function from a sheet to a solid for easy modelling of thin-walled plastic parts.

For more perfect integration of the CAD/CAE system, the features for the feed system including gates and runners should be added to the feature library, and automated medial axis transformation should be implemented to extract a sheet model of the medial surface for the main shape of the part. The feature mapping shell has been developed only for C-FLOW. In the future, the feature mapping shell for various application areas including not only CAE but also CAPP, CAM, etc., will be developed. The expert system to reflect the result of design and analysis can be linked with this system. The constraint-based design functionality can be added to our design module. By expanding the system with the above functions, the integrated CAD/CAE/CAM system to support design and manufacturing of a plastic product and its injection mould will be achieved.

REFERENCES

- [1] Huh, Y. J., and Kim, S. G., "Ribber: A Knowledge-Based Synthesis System for Ribbed Injection Molded Parts", Concurrent Product and Process Design, Edited by Chao, N. H. and Lu, S. C. Y., The Winter Annual Meeting of the ASME, San Francisco, PED-Vol.36, December 10-15, 1989, 195-204.
- [2] Ishii, K., Hornberger, L., Liou, M., "Compatibility-Based Design for Injection Molding", Concurrent Product and Process Design, The Winter Annual Meeting of the ASME, San Francisco, DE-Vol.21, PED-Vol. 36, December 10-15, 1989, 153-160.
- [3] Lee, S. H. and Lee, K., "An Integrated CAD System for Mold Design in Injection Molding Process", Production Engineering Division, The Winter Annual Meeting of the ASME, Chicago, PED-Vol.32, 1988, 257-271.
- [4] Wang, K. K., et.al, "Integration of CAD/CAM for Injection - Molded Plastic Parts", CIMP Progress Report No.13, 1987.
- [5] Hieber, C. A. and Shen, S. F., "A Finite-Element / Finite-Difference Simulation of the Injection-Molding Filling Process", Journal of Non-Newtonian Fluid Mechanics, Vol.7, 1980, 1-32.
- [6] Lee, H. S., Sin, H. C., Kim, S. G., "Integrative Boundary Pressure Reflection Method for Injection Mold Filling", Polymer Engineering and Science, Vol.30, No.23, 1990, 1513-1522.
- [7] "C-FLOW User's Guide", Advanced CAE Technology, Inc., 1989.
- [8] Gadh, R., Hall, M. A., Gursoz, E. L., Prinz, F. B., "Knowledge Driven Manufacturability Analysis from Feature-Based Representations, The Winter Annual Meeting of the ASME, San Francisco, DE-Vol.21, PED-Vol. 36, December 10-15, 1989, 21-34.
- [9] Bernhardt, E. C., "CAE for Injection Molding", Hanser Publisher, New York, 1983.
- [10] Lee, K. and Kwon, B. W., "Efficient Modeling Method of Sheet Objects", Computers in Engineering, Proceedings of the International Computers in Engineering Conference and Exhibit Vol.1, Published by ASME, New York, NY, USA, pp.437 - 446, 1992.
- [11] Shah, J. J., "Assessment of Features Technology", Computer-Aided Design, Vol.23, No.5, June 1991, 331-343.
- [12] Yu, X., Goldak, J. A., Dong, L., "Constructing 3-D Discrete Medial Axis", Proceedings of Symposium on Solid Modeling Foundations and CAD/CAM Applications, 1991, 481-489.
- [13] Weiler, K., "Topological Structures for Geometric Modeling", Ph.D. Thesis, Rensselaer Polytechnic Institute, August 1986.
- [14] Lee, S. H. and Lee, K., "Partial Entity Structure: A Fast and Compact Non-Manifold Boundary Representation Based on Partial Topological Entities", The Sixth ACM Symposium on Solid Modeling and Applications, to appear, June 2001.
- [15] Masuda, H., "Topological operators and Boolean operations for complex-based nonmanifold

geometric models," Computer-Aided Design, Vol. 25, No. 2, February 1992, 119-129.

- [16] Pratt, M. J., "Synthesis of an Optimal Approach to Form Feature Modeling", Computers in Engineering, Proceedings of the 1988 ASME International Computers in Engineering Conference, San Francisco, July 31 - August 14, 1988, 263-274.
- [17] Rosen, D. W., Dixon, J. R., Dong, X., "A Methodology for Conversions of Feature-Based Representations", Design Theory and Methodology, ASME, DE-Vol.31, 1991, 45-51.