Title:
Automatic Update of Feature Model after Direct Modeling Operation

Authors:
Xiaolian Qin, qqz003@163.com, State Key Laboratory of CAD&CG, Zhejiang University
Zhihong Tang, tang_zhi_hong@126.com, State Key Laboratory of CAD&CG, Zhejiang University
Shuming Gao, smgao@cad.zju.edu.cn, State Key Laboratory of CAD&CG, Zhejiang University

Keywords:
Automatic Update, Direct Modeling, Parametric and Feature-Based Modeling, Design Intent Maintenance, CAD

DOI: 10.14733/cadconfP.2020.16-20

Introduction:
Feature-based modeling and direct modeling are the two mainstream modeling methods [1, 5, 7]. As feature-based modeling has a good performance in global design and direct modeling behaves well in local design, abilities of these two methods are complementary to each other. Major engineering CAD vendors has begun to push out direct modeling module in their products, e.g. Creo [2], CATIA [3], NX [6]. However, integration between feature-based modeling and direct modeling is still an open issue. The key technical issue to be resolved is: how to determine the updated feature model after direct modeling operation. The essence of the issue is to keep the consistency between the updated feature model and the new B-rep model. Therefore, a systematic update method of feature model is needed after direct modeling operation. Fu [4] firstly put forward this issue and presented a method based on cellular model. Model validity is guaranteed by applying cellular model, which is rarely used in most of CAD products. Only extrusion features are taken into consideration, lacking its ability to handle more complex models. Zou [8] presented a method of analyzing geometric constraint system of model after direct modeling operation, which is based on pure 3D geometric constraint system. The work focuses on 3D geometric constraint analysis, but no update operations are made to geometric constraint system after analysis.

In order to meet the needs of integration of direct modeling and feature-based modeling, we propose an approach to resolving the problem of updating feature-based model after direct model operation. A program is implemented and several cases are tested to verify the validity of the method.

Main Idea:
The problem is stated as follows:
Problem Given a feature model M_f and its associated B-rep model M_b, direct modeling operations O are executed with a new edited B-rep model M_b' obtained, find the updated feature models consistent with M_b'.

Feature is the basic element of feature model. To determine the updated feature model, the influenced features should be found out firstly. As update strategies of features may not be unique, a candidate set of update operations can be push forward, after which the best one is sorted out to obtain the updated feature model.

Based on analysis above, a resolving algorithm is given in Algorithm. 1. Three critical steps involved are described in detail in the following subsections.

Algorithm. 1: Automatic update of feature model after direct modeling operation.

Input: O, M_b, M_f – Direct modeling operations, original B-rep model, original feature model
Output: M_f' – Updated feature model

1. $V \leftarrow$ DeterminationOfFeatureVolumeVariation (O, M_b, M_f)
2. $C \leftarrow$ GenerateCandicateOperations (V)
3. $A \leftarrow \varphi$ //Array of operation scores
4. **for** each candidate operation $c \in C$ **do**
5. $s \leftarrow$ Evaluate (c)
6. add s to A
7. **end for**
8. $o_o \leftarrow$ GetOptimalOperation (A, C)
9. $M_f' \leftarrow$ UpdateFeatureModel (o_o)
10. **Return** M_f'

Determination of Feature Volume Variations

In order to determinate the updated feature model, the affection of direct modeling operations should be firstly converted to variations of feature volumes. For this purpose, two main issues need to be addressed: a) which features are influenced? b) How do they change?

For the first issue, features influenced are always relative to the push-pulled faces in direct modeling operations. In feature-based modeling, a boundary face in B-rep model is constructed from a set of original feature faces by Boolean operations, hence if the face is transformed, features relative to it are certainly influenced. According to the relation between two feature faces being merging or trimmed, this paper divides the influenced features into **direct relevant features** and **indirect relevant features** (Fig. 1). Each type of relevant feature can be identified according to its characteristics.

Based on the identification result, affection of direct modeling operations can be converted to variations of relevant feature volumes, which are called **feature-variation volumes**. Feature-variation volumes are 3D bounded regions with positive and negative attributes. Additionally, we noted that in some cases, some of variational regions cannot be converted to feature-variation volumes, which are called **independent-variation volumes** (Fig. 2).

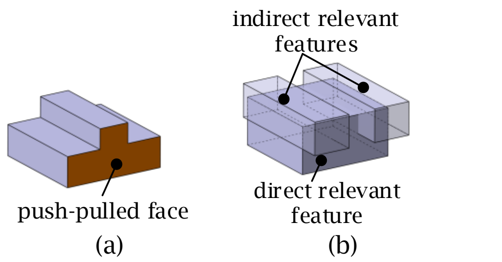


Fig. 1: Influenced features in direct modeling operation. (a) A B-rep model with a push-pulled face selected, (b) Corresponding direct and indirect relevant features.

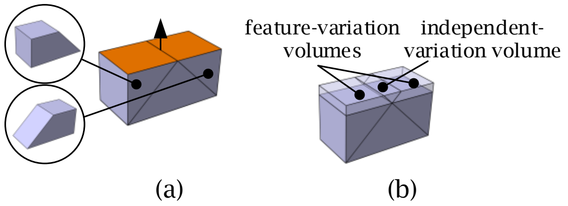


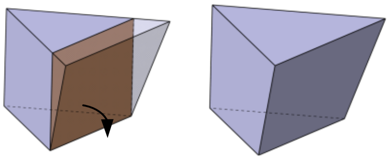
Fig. 2: A case with independent-variation volume. (a) Direct modeling operation to a model with overlapped features, (b) Appearance of independent-variation volume.

Generation of Candidate Update Operations

A satisfying updating way of an influenced feature is converting the feature-variation volume to parameter modification. However, not all kinds of feature-variation volumes can be converted successfully as some of them may result in invalid features, which destroy the inherent semantics (Fig. 3). To resolve this issue, this paper adopts a strategy of generating candidate update operations. In this strategy, conversion from feature-variation volumes to parameter modifications is firstly tried. When the conversion is failed, to keep feature semantic, only part of the feature-variation volume is converted to form up a valid feature. Then, the remaining volumes are to be added as new features. After that, if previous steps cause inconsistency between the updated feature-based model and B-rep model, feature reordering is executed. The critical steps are described in detail as follows.

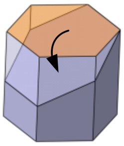
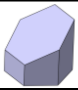
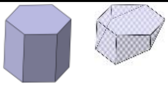
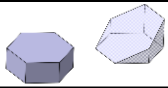
1) Parameter modification

For each influenced feature, we merge the original feature volume and feature-variation volume and give a simple feature recognition to the resulting volume based on the original feature volume. If it fails in feature recognition or the feature has a huge change, a method called **feature mending** is used to mend the feature volume to a valid feature volume. Apparently, another volume will be generated, but such operations can help to keep feature semantic. As is shown in Tab. 1, three candidate operations are listed for a variation of extrusion feature volume.



(a) (b)

Fig. 3: A conversion destroying feature semantic. (a) Variation of an extrusion feature volume, (b) The result is no longer an extrusion feature.

Model variation	*Operations*	*Description*
		Modify extruding plane.
		An extrusion feature volume with a negative volume.
		An extrusion feature volume with a positive volume.

Tab. 1: Candidate update operations of an extrusion feature volume variation.

2) Feature addition

Remaining volumes, including independent-variation volumes and feature-variation volumes after parameter modification, are supposed to be instantiated as new features. Firstly, enumerate all possible strategies to divide the remaining volumes into a set of groups. Volumes in each group are supposed to be adjacent and able to merge into a new one. Secondly, for each strategy, a set of composite volumes are obtained by groups. For each composite volume, if it can be recognized successfully as a general feature, a new parameterized feature is to be instantiated based on the composite volume, otherwise the composite volume is to be instantiated as a user-defined feature. Finally, the new features are added to the end of the feature history in the model.

3) Feature reordering

In feature-based modeling, if a feature *A* is added earlier than feature *B*, *B* will cover *A* in their overlapped region. Hence, after direct modeling operation, the feature-variation volume of a feature may be covered by another feature added later, which sometimes causes inconsistency between the updated feature model and the new B-rep model (Fig. 4). The inconsistency is essentially resulted from unreasonable feature history, so an effective way to resolve it is feature reordering.

The main idea of feature reordering is determining the dislocated features and move them to the right positions in feature history. Firstly, for each feature, its preposition feature set of which features should locate before it is supposed to be found out. A feature *A* is the preposition feature of a feature *B* if it satisfy one of the following conditions: (a) for a push-pulled face, *A* is the direct relevant feature and B is the indirect relevant feature. (b) Feature-variation volume of A has intersection with original feature volume of B and they have inverse attributes. Then, according to current feature history, the dislocate features can be determined. The moving strategy of dislocate features is swapping it with the next feature recursively until a swappable feature is found. At last, if there isn't a feature which has corresponding dislocate features, the inconsistency is eliminated.

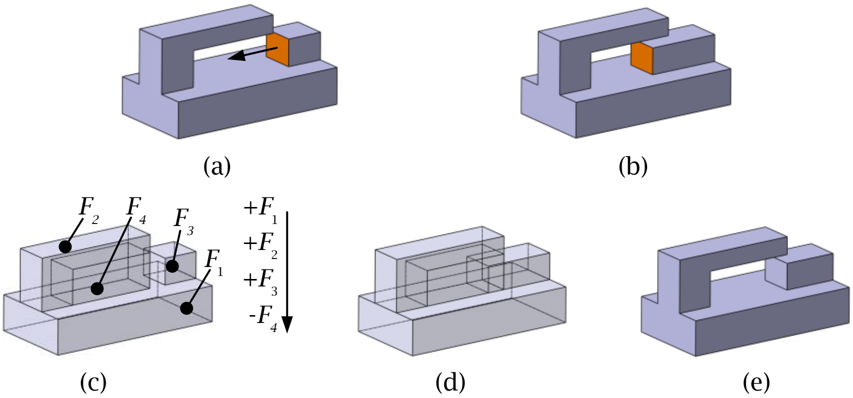


Fig. 4: Inconsistency results from feature order. (a) Original B-rep model and a direct modeling operation, (b) Resulting B-rep model, (c) Original feature model, (d) Updated feature model without feature reordering, (e) Inconsistent B-rep model generated from (d).

Determination of Optimal Update Operation

An optimal update operation is to be found from a set of candidate operations. Basically, a good update operation should maintain feature semantics and user's design intent as much as possible. Therefore, an approach is presented in this paper to determining the optimal update operation based on evaluation.

Penalty function is applied in evaluation of candidate operations. Three aspects are considered in the evaluation as follows: modifications of feature parameters, variations of feature order and additions of new features. Given an original feature model ori and an updated feature model $cand$ from a candidate operation , the penalty function of the operation can be defined as:

$$E(cand, ori) = \omega_m E_m(cand, ori) + \omega_a E_a(cand) + \omega_p E_p(cand, ori) \tag{2.1}$$

In this formula, ω_m , ω_a and ω_p are weights. E_m , E_a and E_p are penalty functions described as follows.

1) E_m : penalty function of feature parameter variations

Penalty value of feature parameter variations can be calculated by accumulation of all the feature parameter variations. Although different types of features have different kinds of parameters, they all have sketch in common. Therefore, penalty function of single feature parameter variation is composed by two parts: sketch variation and other parameter variations. Penalty of sketch variation can be further divided into geometry variation and topology variation.

2) E_a : penalty function of feature additions

The factors influencing penalty value of feature additions come to the number and complexity of new features. The complexity of a new feature is measured according to its geometry complexity and topology complexity.

3) E_p : penalty function of feature order variations

It is relatively simple to define feature order variations. For each feature, we can define its order variation as the position difference before and after in feature history. Penalty value of whole feature order change can be calculated by accumulation of all the absolute position difference value.

Experimental Results

The update method presented above has been implemented. The graphical user interface is shown in Fig. 3. A representative case is presented in Fig. 4. The model shown in Fig. 4 is composed by 19 features. A direct modeling operation is executed on the red face and a new B-rep model is obtained. The program outputs an updated feature in Fig. 4(b) with 3 influenced features changed. The B-rep model generated by updated feature model is consistent with the B-rep model obtained by direct modeling operation.

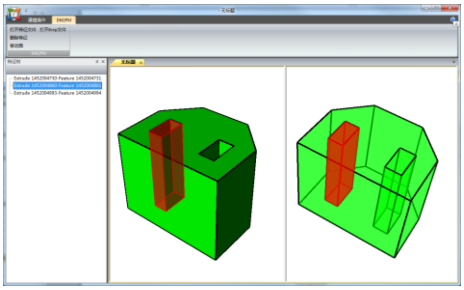
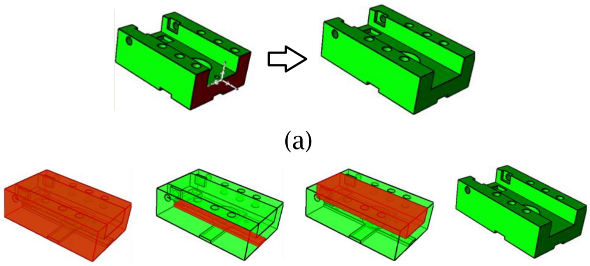


Fig. 3: Graphical user interface.



(a)

(b)

Fig. 4: Case. (a) Direct editing, (b) Optimal update operation.

Conclusions:

A method is presented in this paper to update feature model after direct modeling operation, consisting of determination of feature volume variations, generation of candidate update operations and determination of optimal update operation. The method can successfully generate a new feature model consistent with the new B-rep model after direct modeling operation. Limitations are: (a) only push-pulling of faces are discussed in direct modeling operation. (b) Computation load of updating may become significant when a good number of features are influenced by direct modeling operation.

References:

[1] Ault, H.; Phillips, A.: Direct modeling: easy changes in CAD? Proceedings of the 70th ASEE engineering design graphics division midyear conference. 2016. p. 99–106. https://commons.erau.edu/asee-edgd/conference70/papers-2016/14

[2] Direct Modeling with Creo, https://www.ptc.com/en/products/cad/creo

[3] Direct Modeling with Dassault Systems. http://perspectives.3ds.com/interviews/direct-modeling-with-dassault-systemes

[4] Fu, J.; Chen, X.; Gao, S.-M.: Automatic synchronization of a feature model with direct editing based on cellular model. Computer-Aided Design and Application, 14(5), 2017, 680-692. https://doi.org/10.1080/16864360.2016.1273585

[5] Shah, J. J.; Mantyla, M.: Parametric and Feature-based CAD/CAM: Concepts, Techniques, and Applications. John Wiley & Sons, 1995.

[6] Synchronous technology with NX, http://www.plm.automation.siemens.com

[7] Tornincasa, S.; Monaco, F. D.: The Future and the Evolution of CAD. Proceedings of the 14th international research/expert conference. 2010. p. 11-8. https://doi.org/10.1299/jsdd.4.575

[8] Zou, Q.; Feng, H.-Y.: Variational B-rep model analysis for direct modeling using geometric perturbation. Journal of Computational Design and Engineering, 6(4), 2019, 606-616. https://doi.org/10.1016/j.jcde.2019.03.002