Title:
Automatic Synchronization of a Feature Model with Direct Editing

Authors:
Jun Fu, fujun3@gmail.com, State Key Laboratory of CAD&CG, Zhejiang University
Xiang Chen, xchen@cad.zju.edu.cn, State Key Laboratory of CAD&CG, Zhejiang University
Shuming Gao, smgao@cad.zju.edu.cn, State Key Laboratory of CAD&CG, Zhejiang University,

Keywords:
Direct Modeling, Feature-based Modeling, Automatic Synchronization, Cellular Model, Design Semantics

DOI: 10.14733/cadconfP.2016.242-245

Introduction:
In the last two decades, parametric and feature-based modeling [8] played a great role in computer-aided design and brought huge economic benefits to manufacturing industries. Rather than low-level editing, the technology enables a designer to create or edit high-level design elements, i.e. "*feature*", which possesses rich engineering/manufacturing semantics. However, feature-based modeling has its own drawbacks [1], e.g. high complexity, order dependency and predefinition of changeable parameters, which can hamper the modeling flexibility. Recently, direct modeling technology is rapidly developing and getting used in commercial CAD systems like SpaceClaim [10], NX [9] and Creo [7]. Direct modeling is friendly to normal users, as one can easily manipulate geometric elements to edit a native model, without understanding the complex or even tricky design semantics in the feature model. Compared with feature-based modeling, direct modeling is a lightweight editing tool, being more efficient and flexible. Early/conceptual design or personal customization/fabrication can all benefit from it. While the above two technologies have their own merits, they can be complementary to each other. They are often used in different design phases or preferred by experts from different domains. Actually, the communication between the two modeling methods is crucial to design success, as designers need to switch between different design phases, and domain experts need to share idea or edit models during collaboration. Unfortunately, this kind of communication is still very hard, which severely hinders design innovation or even fails a project.

While feature-based modeling uses feature model to record the parametric design process, direct modeling operation is an immediate event that makes a change to geometric model (B-rep). Once users edit a parametric model through direct modeling, the underlying B-rep is modified without involvement of the features. To keep the parametric information up-to-date and valid, which is imperative for supporting the further feature editing, the feature model should be synchronized accordingly, and the modified geometric model can be faithfully regenerated by the new feature model. This is the key to maintain the consistency between the B-rep updated by direct modeling operations and the B-rep derived from the modified feature model and enable communication between the two modeling technologies.

Main Idea:
In this paper, we present an automatic method to smoothly convert direct modeling operations into feature model modification operations. Based on feature representation harnessed on cellular model [6], we evaluate the influence of geometric editing on the features and update the feature model. It must be emphasized that the feature model modification is not unique, as there are many possible ways to update the feature model for regenerating the latest geometric model. We thus design three

synchronization strategies following the principle of least astonishment: a) modify feature parameters; b) adjust feature orders; and c) add/remove specific features. We executed them sequentially to ensure a successful synchronization, while introducing least modifications to the original feature model.
Our synchronization method takes as input (i) a feature model M including a feature tree, (ii) a direct modeling operation d that modifies the underlying geometries of M. In particular, M is defined as $\{F, O, G\}$, in which $F = \{F_0, F_1, \cdots, F_n\}$ is the sequence of features, $O = \{\otimes_1, \otimes_2, \cdots, \otimes_n\}$ is the sequence of boolean operations "+, − and ∩" applied on features, and G is the geometric model generated by:

$$G = F_0 \otimes_1 F_1 \otimes_2 F_2 \cdots \otimes_n F_n.$$

After conducting the direct modeling operation (Fig. 1(b)), we have a new geometric model $G' = d(G)$.

Our *synchronization* method generates a new feature sequence F' and the corresponding boolean operation sequence O' such that the geometric model computed from F' is exactly the same as the edit result G':

$$F'_0 \otimes'_1 F'_1 \otimes'_2 F'_2 \cdots \otimes'_m F'_m = G'.$$

We note that F' satisfying the above constraint is usually not unique. By following the principle of least modification to the original feature model, whose objective is to make the new feature model generated keep the design intent involved in the original feature model as much as possible, we design the following three synchronization strategies:

1. ***Modification of feature parameters*** Modify the parameters and sketch of the original features.
2. ***Adjustment of feature orders in feature tree*** Adjust the positions of specific features in feature tree while maintaining the inherent feature-dependencies.
3. ***Addition and deletion of features*** Add new features into feature tree or delete useless features from it, which ensures the success of synchronization.

In view that the influences of the above three strategies on the feature model are increasingly expanded with respect to the change on the design intent, it should be in general reasonable to conduct the above three strategies sequentially. As shown in Fig. 1(c), the three strategies are sequentially applied to generate a valid F' that preserves the design semantics as much as possible.

Algorithm 1 lists the main pipeline of our synchronization processing. The three synchronization strategies are executed in order, until the underlying geometry of the new feature model is consistent with the direct editing result.

Algorithm 1 Feature Model Synchronization

```
procedure SYNCHRONIZATION(M, G')
  CM ← ConstructCellularModel(M)
  CM.AddFeature(G')
  if IsConsistent(CM) then return
  ModifyFeatureParameters(CM)
  if IsConsistent(CM) then return
  AdjustFeatureOrders(CM)
  if IsConsistent(CM) then return
  AddDeleteFeatures(CM)
  return
```

Proceedings of CAD'16, Vancouver, Canada, June 27-29, 2016, 242-245

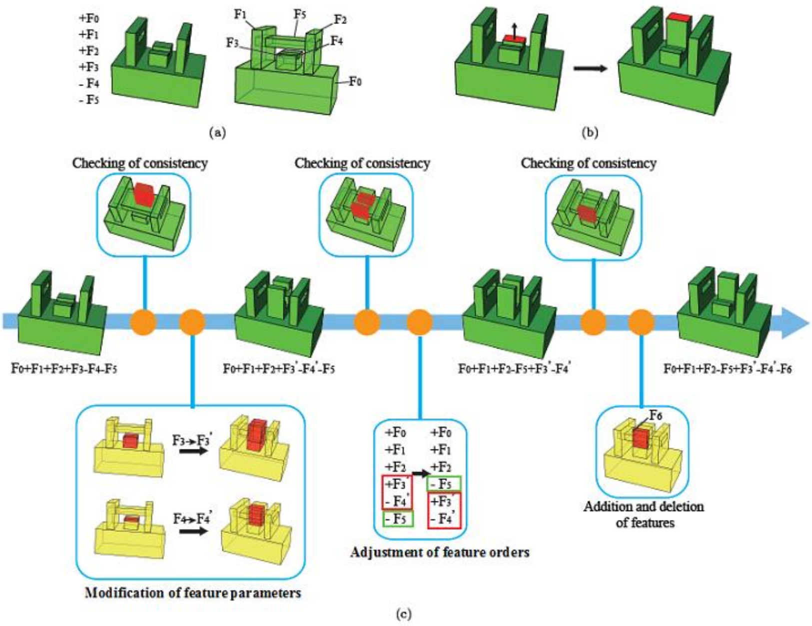

Fig.1: Overview of synchronization method: (a) the original feature model; (b) the direct editing; (c) the synchronization process.

Conclusion:

In this paper, we proposed a novel synchronization method to automatically keep the feature model consistent with the direct editing result, forming a seamless bridge between direct modeling and feature-based modeling. By successively executing necessary modifications of features, adjustments of feature orders and additions/deletions of features, the method not only ensures a successful synchronization solution, but also preserves the design semantics of original model as much as possible. During the process, we leverage the cellular model to efficiently and effectively check geometric consistencies and manipulate features. The experimental results show that the synchronization algorithm works well for models with extrusion features. In all, we believe our method opens the possibility towards an elegant fusion between feature-based modeling and direct modeling technologies.

In the future, we plan to study the method of determining the optimal modification to the feature model after direct modeling operations, and explore the support of geometric constraints during synchronization, which largely enriches the design semantics of feature model and hence requires more complex algorithms. In addition, we are also interested in synchronization of non-extrusion features and even assembly model.

Acknowledgements:

The authors are very grateful to the financial supports from NSF of China (61173125) and National 863 High Technology Project (2013AA041301).

References:

[1] Bidarra, R.; Bronsvoort, W. F.: Semantic feature modelling, Computer-Aided Design, 32(3), 2000, 201–225. http://dx.doi.org/10.1016/S0010-4485(99)00090-1

[2] Bidarra, R.; Kraker, K. J. D.; Bronsvoort W. F.: Representation and management of feature information in a cellular model, Computer-Aided Design, 30(4), 1998, 301–313. http://dx.doi.org/10.1016/S0010-4485(97)00070-5

[3] Bidarra, R.; Madeira, J.; Neels, W. J.; B. W. F.: Efficiency of boundary evaluation for a cellular model, Computer-Aided Design, 37(12), 2005, 1266–1284. http://dx.doi.org/10.1016/j.cad.2004.12.006

[4] Cellular topology data structure, http://doc.spatial.com/index.php/Cellular_Topology_Data_Structure

[5] Direct modeling - who and why needs it? a review of competitive technologies , http://isicad.net/articles.php?article_num=14805

[6] Parametric or direct modeling: why you may need both , http://www.econocap.dk/assets/files/creoparametric/6275_Parametric_Direct_CAD_WP.pdf

[7] Ptc Creo Elements/Direct Modeling, http://www.ptc.com/product/creo-elements-direct/modeling

[8] Shah, J. J.: Parametric and feature-based CAD/CAM: concepts, techniques, and applications. John Wiley & Sons, 1995.

[9] Siemens corporation, http://www.plm.automation.siemens.com/en_us/products/nx/

[10] Spaceclaim corporation, http://www.spaceclaim.com/en/

[11] Spaceclaim in manufacturing , http://partnerfiles.spaceclaim.com/Collateral/White%20papers/Develop3D_SpaceClaim_in_MFG.pdf

[12] Synchronous technology, http://www.plm.automation.siemens.com/legacy/docs/SynchronousTechnology_CPDA_WhitePaper.pdf