Proceedings of the Twelfth National Conference on Manufacturing Research, *Advances in Manufacturing Technology X*, eds. A N Bramley, A R Mileham and G W Owen, University of Bath, pp256-260, September 1996

Representation Validation in Feature-Based Modelling: A Framework For Design Correctness Analysis and Assurance

M S Hounsell & K Case
Manufacturing Engineering Department, Loughborough University
Loughborough, LEICS, LE113TU

Summary

Feature-based Modelling allows extra meaning to be added to geometry, but lacks the equivalent geometric formalism usually found in computer-aided design (CAD) and Geometric Solid Modelling (GSM) systems. CAD systems have been evolving into constraint-based design environments instead of intent-driven ones where the designer can use whatever manipulation is available in the system without been afraid of messages like "manipulation not permitted". These messages usually restrain the user in order to avoid representation changes and faulty or "unknown" situations.

A Design-by-Feature system with a representation validation framework is presented that supports "Design for X", intent-driven modelling, encompasses existing low-level geometric verifications, adds high-level rules to analyse and enrich the design and incorporates operations to assure its correctness. Also it alleviates the designer from specifying each and every geometric detail/relationship (improving productivity).

1. Introduction

Features play an important role in capturing the designer's intent in computer-aided design (**CAD**) raising the abstraction level of geometric design and facilitating integration with applications such as computer-aided manufacturing (**CAM**) and computer-aided process planning (**CAPP**). However, such integration will only be profitable if the feature model is valid in terms of the roles played by features. For instance, if a 'pocket' (or a 'blind hole') in the model is allowed to pass through the part, this misrepresentation could cause machine damage, mistakes or, at least, non-optimised decisions by a CAPP system.

Model validation seeks to prove that the model does its job in a variety of circumstances and that the model agrees with the "real thing". Feature-based Modelling (**FBM**) has already been accepted as a 'valid (and indeed, necessary) modelling framework to represent artefacts' geometric design to the next generation of more intelligent CAD and Geometric Solid Modelling (**GSM**) systems ([4]). Most often there are built-in restraints that apply to the model to guarantee that valid models are within its representation domain and thus this is called *representation validation.*

2. Validation Interpretations

Some validation implementations perform *ad-hoc* specific computations or (even worse) tend to over-constrain the user's manipulation capability ([3], [9], [15]). Others perform validations verifying if one type of high-level feature representation implementation (such as volumetric, implicit or intentional features) have a 'proper' and valid GSM evaluated counterpart ([1], [5], [11]). Those that include parametric constraining at the feature level implement validation by solving feature parameter conflicts ([2]. [6], [7]. [8], [14]).

Most of the validation concepts found in the literature are basically geometric-driven. The literature shows that a lack of attention has been paid to the validation of a given feature-based representation. However, it has been suggested that 'another layer of validation becomes necessary' ([10]) beyond the geometric one: the conceptual (semantic) validation.

GSM systems have two basic levels of validity conditions: organisational (topological) and structural (geometrical). Considering that the majority of feature modellers are integrated in some way to an underlying GSM system, it can be seen that both sets of GSM validity conditions play the role of a structural validity level for Feature Validation.

The other organisational validity conditions are termed *Conceptual Validity Conditions* because they are concerned with the feature's *concept* (their role/semantics as a 3D modelling technique, their expected behaviour and, their high-level organisational meanings). Conceptual Feature Validation thus implies that the verification of the intended functionality of a given feature must conform with the geometric semantic meaning *(designer's intent)* assigned to that specific feature type.

3. Feature Validation Framework

Validating a feature-based representation is a very subjective and 'difficult problem to handle in the most general sense' ([12]) and depends heavily on the role the feature plays with respect to a particular application. It is a 'very difficult and obscure task because features themselves are not well understood with their extra meaning, purpose and objectives in addition to the embedded geometric data representation' ([11]).

Although 'there are no universally applicable methods for checking the validity of features' ([13]) to perform the representation validation three elements were identified as necessary to compose a Feature Validation System (FVS): (i) the domain characterisation, (ii) the set of checking procedures and, for practical reasons, (iii) operators to transform invalid representations (into valid ones).

The extra formalism to be added to the characterisation of feature's paradigm should establish a set of validity conditions according to a specific interpretation of the features.

The representation is then tested against these conditions and if it proves to be invalid, revalidation operators could be used to tum it into a valid representation.

The very definition of features and their characterisation should be made in such a way as to be suitable for verification and to be in accordance with expected common sense behaviour of features. A volumetric analysis of features seems to be an adequate (feasible) candidate for this purpose.

Further, if the characterisation formalism is made clear, verifiable and representative enough then the system could perform automatically the identification of complex relationships between features. This automatic recognition will promote the designer's freedom from this tedious task and will enrich the representation. However, the human understanding of the model and her/his intervention will be *necessary* to accept or reject the recognised relationship as an important and desired one. And, once these relationships are meaningful as well as features are, this process will drive a more conversational user interface.

The feature-based reasoning should mainly use feature types, descriptions and parameters (rather than their geometrical Brep or CSG evaluations) as a 'vocabulary' for validation analysis, manipulation and revalidation operations. Furthermore, some engineering requirements (such as manufacturability) could be defined using this vocabulary and embedded into the validity condition set of a FVS allowing "Design for X" strategies to be achieved. Note that "X" could stand for machinability, assemblability, fixturing or recycling, as an example.

4. At last, a FRIEND

A prototype system called **FRIEND** (an acronym for **F**eature-based **R**easoning system for **I**ntent-driven **EN**gineering **D**esign) is being implemented adopting the feature validation framework presented here. Conceptual Feature Validity Conditions in **FRIEND** are translated as reasonings and enquiries to the underlying GSM but mainly to information stored in the Feature Modeller.

Feature domain characterisation in **FRIEND** has been divided into two types: *properties* and, *intents.* Some characterisation *properties* were found to resemble GSM properties and are based on a volumetric interpretation of features. This resemblance emphasises the common-sense behaviour expected from features. Besides the volumetrical and behavioural properties assigned to features, a full characterisation of the feature's domain should be established regarding its intents (functional, structural and geometrical, for instance). With such a thorough characterisation established, an intent-driven environment is achieved via the FVS framework presented above.

Intent-driven geometric design in **FRIEND** is an approach where intents are pre-defined by the user and maintained (and even identified) by the system. It can encompass parametric dependency constraining (which establishes and guarantees numeric relationships between dimensions/parameters) but, extrapolates this idea and makes it more flexible by allowing a greater variety of relationships between (parts of) the features to be used. *Intents* are much more broad in spectrum, change dynamically, and are powerful in representing engineering aspects. This gives more freedom to the designer than environments with only parametric constraints. Intents analysis and definition in the FBM domain are a complex topics and are outside the scope of this paper.

5. Conclusion

A gap in the validation of CAD's feature-based representations has been identified and a framework to overcome this drawback is presented. This framework results in the identification of three components of a Feature Validation System (**FVS**): domain characterisation, validity conditions and revalidation operations. Such a sub-system embedded within a Feature Modeller could perform complex "Design for X" analysis in an intent-driven environment once an adequate characterisation of the domain is available.

The reasoning process is basically a rule-based one and uses mainly features (and their data-structures) as a vocabulary rather than GSM data. Automatic recognition of pre-established designer's intents and the consequent representation enrichment can be achieved which raises the quality and usefulness of the model as well as relieving the designer of these tedious tasks. An FVS framework is easily integrated into GSM systems because the feature's concept is kept separate from its low-level GSM counter-part evaluations.

The concepts mentioned here - FVS framework, feature-based reasoning and, intent-driven design - were adopted in the development of a feature modeller called **FRIEND** (an acronym for **F**eature-based **R**easoning system for **I**ntent-driven **EN**gineering **D**esign).

6. References

l. Bidarra, R. and Teixeira, J.C. *"A Semantic Framework for Flexible Feature Validit*y Specification and Assessment". (ASME) Computers in Engineering, vol **1,** pp 151-158. 1994

2. Bronsvoort, W.F. and Jansen, F.W. *"Feature Modelling and Conversion- Key Concepts to Concurrent Engineering".* Computers in Industry, vol **21**, pp 61-86. 1993.

3. Dixon, J .R. and Cunningham , J.J. *"Research in Designing with Features"* . IFIP WG 5.2, Conf. on Intelligent CAD, Boston, vol **l**, pp 137-148. 1987.

4. Dixon, J.R., Libardi Jr, E.C., and Nielsen E. H. *"Unresolved Research Issues in Development of Design-with-Features Systems".* IFIP WG 5.2 NSF Working Conf. on Geometric Modelling, ResselaerVille - USA, Elsevier Sci. Pub., vol **l**, pp 183-196. 1990.

5. Duan, W ., Zhou, J., and Lai, K. "*FMST: A Feature Solid Modelling Tool for Feature-Based Design and Manufacture".* Computer Aided Design, vo1 **25**(1), pp 29-38. 1993.

6. Emmerik, M.J.G.M. van. *"Interactive Design of 3D Models with Geometric Constraints".* Visual Computer, vol **7**(5-6), pp 309-325. 1991.

7. Nielsen, E.H., Dixon, J.R., and Zinsmeister, E.E. *"Capturing and Using Designer Intent in a Design - with-Features System".* DTM'91: 3rd. Int. Conf. on Design Theory and Methodology (ASME-DE), vol **31**, pp 95-102. 1991.

8. Ovtcharova, J. and Jasnoch, U. *"An Integration of Feature-Based Design and Consistency Management in CAD Management".* IFIP Intl. Conf. on Feature Modelling and Recognition in Advanced CAD/CAM Systems, Valenciennes, May, vol **2**, pp 739-756. 1994.

9. Pratt, M.J. and Wilson, P.R. *"Requirements for Support of Form Features in a Solid Modelling System".* CAM-I Inc. Report R-85-ASPP-01, Arlington, Texas, USA, 1985.

10. Pratt, M.J. *"Application of Features Recognition in the Product Life-Cycle".* International Journal of Computer Integrated Manufacturing, vol **6**(1-2), pp 13-19. 1993.

11. Rossignac, J .R. *" Issues on Feature-based Editing and Interrogation of Solid Models".* Computers and Graphics, vol **14**(2), pp 149-172. 1990.

12. Salomons, O.W., Houten , F.J .A.M. van, and Kals, H.J.J. *"Review of Research in Feature-Based Design".* Journal of Manufacturing Systems, vol **12**(2), pp 113-132. 1993.

13. Shah, J.J. *"Assessment of Feature Technology".* Computer Aided Design, vol **23**(5), pp 331-343. 1991.

14. Sheu, L.-C. and Tin. J. T. *"Representation Scheme for Defining and Operating Form Features".* Computer Aided Design, vol **25**(6), pp 333-347. 1983.

15. Stroud, I. *"Modelling Techniques for Handling Non-Geometric Information".* Second Symposium on Solid Modelling and Applications, May 19-2 1, Montreal. Canada, pp 367- 376. 1993.