Title:
Integration of Negative Knowledge into MCAD Education to Support Competency Development for Product Design

Authors:
Harald E. Otto, h.e.otto@univpm.it, Polytechnic University of Marche
Ferruccio Mandorli, f.mandorli@univpm.it, Polytechnic University of Marche

Keywords:
Negative Expertise, Learning from Errors, Transversal Abilities, Curriculum Design, Hybrid Geometric Modeling, Shape Engineering

DOI: 10.14733/cadconfP.2016.44-48

Introduction:
Nowadays, the processes of product design and development are widely supported by using virtual prototypes (VP). The reasons for employing virtual prototypes during product design and development are manifold: for example to benefit from a digital model which supports both the assessment of design concepts and the evaluation of design alternatives, or to perform tests and analysis. Various definitions of VP have been proposed in the past. Within the context of the work presented, we will define a virtual prototype as a "computer simulation of a physical product that can be presented, analyzed, and tested from relevant product life-cycle aspects such as design/engineering, manufacturing, service, and recycling just like a real physical model" ([10], p. 233).

Depending on the specific application domain, a virtual prototype can be constructed by integrating various types of digital model such as finite element method (FEM) models, and computer-aided manufacturing (CAM) models. In such a scenario, most commonly a three-dimensional solid model represents the core of a virtual prototype, and this is usually created by means of a computer-aided design system for mechanical engineering (an *MCAD system*). Since the early stages of its development, three-dimensional geometric modeling has been sub-divided into the domains of solid modeling and surface modeling. Nowadays, however, with rapid developments in MCAD technology, a precise distinction between solid modeling systems and surface modeling systems is no longer viable. Instead, almost all commercially available geometric modeling systems are capable of representing and handling both solid models and surface models. Here, the possible use of an integrated solid/surface model is of particular interest for engineering application domains such as injection molding and blow molding and their related tool design. In such domains, surface models are employed for both the representation of complex shapes and the definition of the separation surface of a mold or the mating surface for a fixture. Solid models, on the other hand, are employed for efficiently performing tasks related to shape engineering, which will eventually convert the skin of an object subject to design into the final solid object subject to manufacturing. Under those circumstances, modern MCAD systems can no longer be classified as either true solid modeling systems or true surface modeling systems, but rather as so-called *hybrid modeling systems*, which are capable of representing and handling, within an integrated framework, connected and/or disconnected sets of surfaces, together with multi-body solids.

Modern MCAD systems provide commands to support both surface and solid modeling, along with the interoperability commands required, for example, for cutting a solid with a surface, converting a surface into a bulk solid or a sheet solid, or copying or offsetting the face of a solid. However, techniques and strategies aimed at creating the solid part and the surface part of the hybrid model still remain very different. They are affected by constraints imposed by the considerable differences that exist in the representation of surface models and solid models. This is a situation, which is also reflected in the structure and requirements of current MCAD system user interfaces and their respective user-system interactions. These demand that users possess certain skills and competencies in order to be able to actually produce usable models. From an educational point of view, teaching how to create and manage usable hybrid models poses new challenges which require a teaching strategy that is beyond the mere explanation of the sum of surface modeling and solid modeling techniques, and their best practice and guidelines. In particular, students have to develop an understanding and working knowledge of the deficiencies that can turn a geometric model into an unusable model, along with forming an awareness of the particular design and modeling situations, which may lead to the introduction of such deficiencies.

Motivation and Objectives:

In the context as outlined, a hybrid model is defined as a geometric model composed of one or more solid bodies and a set of connected or disconnected surfaces. A hybrid modeling environment must be capable of providing both a consistent representation of hybrid models and interoperability commands to execute tasks aimed at, for example, cutting solids by means of using surfaces and adding thickness to a surface to convert it into a sheet solid. The capability to represent and manipulate hybrid models, as provided by modern MCAD systems, permits a more efficient approach to design issues. Prior to the advent of hybrid modeling, such problems were cumbersome to solve as they involved the employment of individual surface and solid modeling environments. Concrete examples from industrial practice of what can currently be achieved include the definition of separation surfaces of an aesthetic shape, conducting the shape engineering of an aesthetic shell, modeling cavities for blow or mold injection processes, and the definition of the opening surface of a mold.

However, the correct use of a hybrid modeling environment requires a user to be aware of the geometrical and topological constraints that models must meet in order to become and remain valid and usable during the entire computer-aided design and development process. Those constraints are not always evident to the user, nor are they automatically controlled by any modeling system. Representative cases relating to such constraints include the correct "sewing" of adjacent patches in a surface model, correct orientation of geometric entities, adequate value setting for a minimum curvature radius and, in general, the definition of a two-manifold surface, which is a basic prerequisite for converting any surface model into a manifold solid model. Although most modern MCAD systems provide commands for inspection and analysis to allow for an assessment of the geometry and topology related quality of CAD models, these commands are not always easy to use nor fully understood by users without a theoretical background in geometric modeling. This is an unfortunate situation, which is due, in part, to the fact that these constraints are closely related to both the mathematical definition of the models and the representation schema that is used to implement the geometric kernel of a modeling system.

From an educational point of view, this situation represents a serious challenge, especially within the current MCAD course for engineering students, offered by the department represented by the authors, where teaching is required to an audience that lacks both a mathematical background in geometric modeling and specific knowledge about computer science and digital model representation. Currently, the traditional approach to MCAD teaching is based mainly on the use of tutorials and practical examples, along with definitions of guidelines and best practice. One advantage of such an approach is the immediate focus on the functional aspect of commands provided by a system and on specific modeling strategies, usually applied to example cases resulting in success. However, this "positive" *what-to-do* approach does not provide sufficient knowledge and know-how about how to deal with unexpected or unusual situations. These are not included in the smooth modeling path that is usually shown during modeling exercises, but they are particularly critical from the point of view of defining usable hybrid models. In order to help students develop awareness of issues deriving from shortcomings and mistakes related to sub-optimal goal settings and inappropriate modeling strategies, and know-how on avoiding getting stuck in undesirable situations during problem-solving, we propose to integrate the traditional "positive" teaching approach with a newly developed approach, which is based on the use of negative knowledge, that is, knowing what not to do in a certain situation, and which assumptions are wrong with regard to a certain problem, and learning from errors.

Background and Scope:

One central element of professional expertise and competency is the ability to recognize and correctly interpret situation-related cues signaling a possible deviation from chosen goals and to avoid grave mistakes. This domain-specific situation awareness and explicit knowledge about what to do in order to avoid errors has been attributed to what is called *negative knowledge*. Several concepts for the theory of negative knowledge have been developed in the various fields of cognitive and computer science, education, and business philosophy.

Early work on the concept of negative knowledge is reported in [4] within the context of negative expertise, where negative knowledge is seen as an element of practical know-how relating to tasks and activities that are specific to a professional domain. This view is related to the experiential knowledge of experts about what can go wrong in their domain of expertise and what kind of strategies may lead to inefficient approaches and sub-optimal results during problem solving, and are therefore better avoided. In [5,6] negative knowledge is discussed within the practice of error culture and is assumed to be a kind of meta-knowledge relating to information about false facts and inappropriate action strategies. Here, employing a contrastive approach, negative knowledge is defined as knowledge about what something is not and how something does not work, which strategies do not lead to solutions, and why certain connections do not add up. Within this concept, practical experience is assumed to be one prominent aspect of negative knowledge development. This position was further advanced in the context of professional learning and expertise as reported in [1,2]. In [8] the authors focus on the declarative aspect of negative knowledge, the *knowing what not to know*, which is in contrast to the by nature more procedural aspect of *knowing what (not) to do*. In their work they distinguish two types of not-knowing relating to the informed and uninformed methods of an individual lacking knowledge relevant to expertise. This distinction addresses in the former case an awareness by the individual of his lack of relevant knowledge, while the latter case supposes both a lack of relevant knowledge and a lack of awareness of this very fact (see also competence levels in [3]). More details on the declarative and procedural aspects of negative knowledge can be found in [1]. This work discusses relationships with meta-cognition, and the epistemic potential to enable new insights into various knowledge-related and learning-related fields (see also discussion in [12]). It also considers the support given to improving certainty in how to proceed in a task, to increasing efficiency during performance, and to enhancing the depth and quality of reflection on actions and performance.

In the framework developed for the approach that is outlined in the following section, negative knowledge is conceptualized, from a theoretical point of view, through relationships to desirable situations, which in turn indicate what is considered a good model (configuration) within a given context employing normative knowledge. Here, desirable situations represent a reduced set of all possible situations (desirable/undesirable). In this scenario, the nature of similarity of desirable situations is determined by reducing variety (cf. [7]), which in turn is realized by avoiding undesirable situations by means of restricting actions that have a high tendency (according to what we know and believe to be true) to lead to them. Hence, negative knowledge in terms of knowing what not to do in a certain situation can be conceptualized as a form of action constraint.

Approach and Implementation:

The design and realization of a novel teaching approach, which integrates negative knowledge as one crucial element within current MCAD education (see Fig.1), have been approached by addressing concept mapping, framework development and implementation as follows. First, in order to define what constitutes an error or mistake to be avoided in respect to a particular situation and the quality of a CAD model, some elements of the concept of negative knowledge as outlined earlier have been mapped to the concept of geometric entity deficiency. This concept is used as a qualitative measure to help express certain characteristics of situations during modeling. These characteristics usually lead to models being poorly structured and are thus better avoided. In other words, deficiency represents the loss of one or several characteristics of a geometric entity meaningful in a certain application domain.

Proceedings of CAD'16, Vancouver, Canada, June 27-29, 2016, 44-48

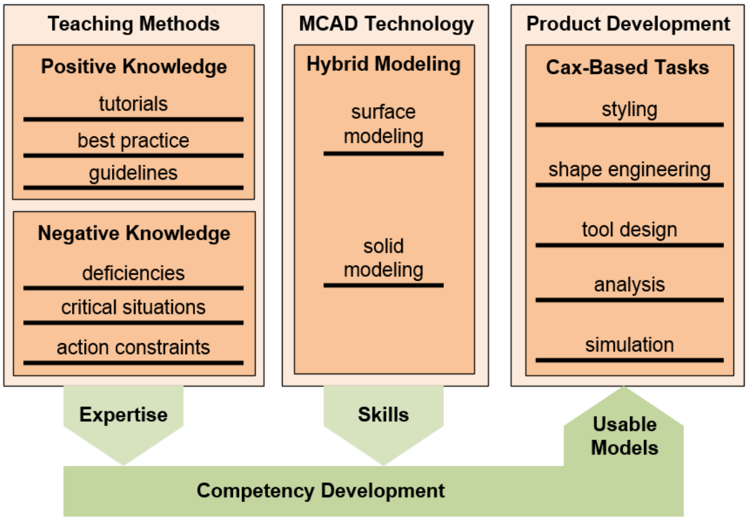

Fig. 1: Overview of teaching methods and hybrid modeling skills related to competency development for product design and development.

Currently, geometric entity deficiencies are sub-divided into groups for single entities and compound entities with the former relating to curves, patches, and solids, while the latter relate to polycurves and polysurfaces.

Second, by formulating negative knowledge as an element of strategic knowledge constraining actions within critical situations that would otherwise lead to errors and mistakes, the concept of a situation can be abstracted as a set of relations associated with particular sets of model configurations, action constraints, anticipated failures, and individual goals and sub-goals. Of particular interest are significant model configurations, which describe a model configuration in a certain context that is significant in respect to action constraints, which in turn are associated with individual actions. These significant model configurations can be related through a mapping to concrete constraints limiting the actions possible in a particular situation. Note that properties that define the quality of the configuration, i.e. whether the model is good or not, are related to the normative knowledge of an application domain.

Third, to implement the approach and integrate it into the current MCAD course in addition to traditional lectures and tutorials, various modeling exercises are provided, which are individually designed for different learning aspects. Results of the exercises are collected and assessed, to identify shortcomings and errors, which usually remain hidden from students due to their limited domain knowledge and expertise. Representative examples of the assessed exercises are later used to discuss, during lectures and also online through the MCAD course web site, issues relating to critical modeling situations, model deficiencies and how to prevent them or initiate a recovery, if feasible. In order to manage the interaction between faculty and students, including the administration of procedures and deadlines for the distribution and collection of exercise material, a web site for the course has been developed within the e-learning platform of the institution's engineering faculty using Moodle, an open source learning management system (LMS, see also [9, 11]).

Conclusions:

The framework and issues regarding current implementation of a novel approach aimed at supporting competency development for product design have been outlined and discussed. As became evident during both theoretical analysis and first empirical work, current efforts to integrate traditional teaching methods with an educational approach based on negative knowledge are indeed capable of supporting capacity development in the form of explicit knowledge about what not to do and a domain-specific situation awareness related to knowledge-based error anticipation.

Promising results were reflected in, among other outcomes, students showing a better overall understanding of issues related to the usability of CAD models and an increased capability to recognize critical modeling situations and thus prevent mistakes typically made by novices. Also, an overall qualitative improvement in planning and performance was observed during problem-solving course work and laboratory exercises, which was achieved, in part, by identifying and subsequently avoiding sub-optimal and inappropriate modeling strategies. To further understanding and insight on both a theoretical and a practical basis regarding the design and implementation of the novel educational approach as outlined, current efforts in the collection of empirical data will be continued. Presentation of detailed results is planned for a forthcoming publication, including statistical details of the examination and analysis of empirical data collected up until the end of the current course.

References:

[1] Gartmeier, M.; Bauer, J.; Gruber, H.; Heid, H.: Negative Knowledge: Understanding Professional Learning, Vocations and Learning, 1(2), 2008, 87-103. http://dx.doi.org/10.1007/s12186-008-9006-1

[2] Gartmeier, M.; Schüttelkopf, E. M.: Tracing Outcomes of Learning from Errors on the Level of Knowledge, in: Bauer, J. and Harteis, C. (eds.), Human Fallibility: The Ambiguity of Errors for Work and Learning, Springer, Dordrecht, The Netherlands, 2012, 33-51. http://dx.doi.org/10.1007/978-90-481-3941-5_3

[3] Heid, H.: Erfordernis und Problematik einer Unterscheidung zwischen Verhalten und Verhaltensdisposition, in: Beck, K., Müller, W., Deißinger, T. and Zimmermann, M. (eds.), Berufserziehung im Umbruch, Deutscher Studienverlag, Weinheim, Germany, 1996, 79-85.

[4] Minsky, M.: Negative Expertise, International Journal of Expert Systems, 7(1), 1994, 13-19.

[5] Oser, F.; Spychiger, M.: Lernen ist schmerzhaft: Zur Theorie des negativen Wissens und zur Praxis der Fehlerkultur, Beltz, Weinheim, Germany, 2005.

[6] Oser, F. K.; Näpflin, C.; Hofer, C.; Aerni, P.: Towards a Theory of Negative Knowledge (NK): Almost-Mistakes as Drivers of Episodic Memory Applications, in: Bauer, J. and Harteis, C. (eds.), Human Fallibility: The Ambiguity of Errors for Work and Learning, Springer, Dordrecht, The Netherlands, 2012, 53-70. http://dx.doi.org/10.1007/978-90-481-3941-5_4

[7] Otto, H. E.; Mandorli, F.: Negative Knowledge and Eco-Design, In Proceedings of the International Conference on EcoBalance, November 20-23, Yokohama, Japan, Paper-No. P-133, 2012.

[8] Parviainen, J.; Eriksson, M.: Negative Knowledge, Expertise and Organisations, International Journal of Management Concepts and Philosophy, 2(2), 2006, 140-153. http://dx.doi.org/10.1504/IJMCP.2006.010265

[9] Walker, R.; Voce, J.; Nicholls, J.; Swift, E.; Ahmed, J.; Horrigan, S.; Vincent, P.: 2014 Survey of Technology Enhanced Learning for Higher Education in the UK, Universities and Colleges Information Systems Association (UCISA), University of Oxford, Oxford, UK, 2014.

[10] Wang, G. G.: Definition and Review of Virtual Prototyping, Journal of Computing and Information Science in Engineering, 2(3), 2002, 232-236. http://dx.doi.org/10.1115//1.1526508

[11] Watson, W. R.; Watson, S. L.: An Argument for Clarity: What are Learning Management Systems, What are They Not, and What Should They Become?, TechTrends, 51(2), 2007, 28-34. http://dx.doi.org/10.1007/s11528-007-0023-y

[12] Winch, C.: Know-How and Knowledge in the Professional Curriculum, in: Young, M. and Muller, J. (eds.), Knowledge, Expertise and the Professions, Routledge, New York, NY, USA, 2014, 47-60.