Available online at www.sciencedirect.com
ScienceDirect
Procedia CIRP 50 (2016) 484 – 489
www.elsevier.com/locate/procedia
26th CIRP Design Conference

A practical methodology for creating robust parametric surface-based models in automotive engineering

Michal Forrai[a], Jana Gavačová[a,*], Ladislav Gulan[a]

[a]*Faculty of Mechanical Engineering, Slovak University of Technology in Bratislava, Námestie slobody 17, 812 31 Bratislava, Slovakia*

* Corresponding author. Tel.: +421-2-5729-6860. *E-mail address:* jana.gavacova@stuba.sk

Abstract

The paper deals with the general description of Generative Engineering Design (GED) methodology used during the development of surface-based components. The stages of GED presented here are applied on sport vehicle parts with visible surfaces. The focus is on the later stages, i.e. detail design. These include creation of robust parametric shape-based CAD model, which is capable of easy change propagation. The GED methodology builds on Knowledge-Based Engineering (KBE) principles, because the ability of quick and easy change is essential. The practical use of the methodology is demonstrated using CAD-based generic templates and a custom tool in CATIA.

Peer-review under responsibility of the organizing committee of the 26th CIRP Design Conference
Keywords: Generative Engineering Design; GED; Knowledge-Based Engineering; KBE; Shape Design; Power Copy

1. Introduction

Product development methodology has been improving for decades. There has been room for improvement in several fields, including CAD applications. Developed methods of designing are relevant and effective only for the complex products. Development of vehicle prototype is a suitable application for the Generative Engineering Design methodology. Car body components with visible surfaces are made from wide range of materials and could have wide range of shapes [1].

Nomenclature

API	Application Programming Interface
C^x	Curve continuity of x-th order
CAD	Computer-Aided Design
CAE	Computer-Aided Engineering
CAM	Computer-Aided Manufacturing
CAS	Computer-Aided Style
COP	Cloud of points
GED	Generative Engineering Design
GSD	Generative Shape Design
GUI	Graphic User Interface
KBE	Knowledge-Based Engineering
VBA	Visual Basic for Applications

The paper consists of two sections, a theoretical and a practical one. Chapter 2 gives a theoretical overview of Generative Engineering Method. Here, relations with other research areas are described (in particular KBE), together with information on GED, its steps and nomenclature. An application example on a vehicle component is given. Chapter 3 is practically focused. A very brief description of relevant general CAD characteristics is given, it then deals with specific implementation details of GED in CATIA. Chapter 3.3 is about further improvement of CATIA capabilities by programming custom software tools using easily accessible tools.

The naming convention used in this paper for general product structure is following (from high-level to low-level elements): product (assembly), part (component), geometrical set, geometrical feature, geometrical element, parameter.

2. General description of GED methodology used during the development of shaped components

The presented approach is related to generative design known in other fields of applied research. Firstly it occurred in the architecture and artistic fields. But later, its advantages were exploited in other appropriate areas [2]. The procedure described here refers to Generative Engineering Design methodology of shaped components. It can be characterized as the ability to generate shape features and their sets through input and output data with evaluation of their relevance in a closed loop. This procedure allows realization of dynamical solution, which is able to absorb and share information back to this process, so that further optimisation can be accomplished. For Generative Engineering Design described in this paper, it is possible to abstract the following definition:

Peer-review under responsibility of the organizing committee of the 26th CIRP Design Conference
doi:10.1016/j.procir.2016.04.145

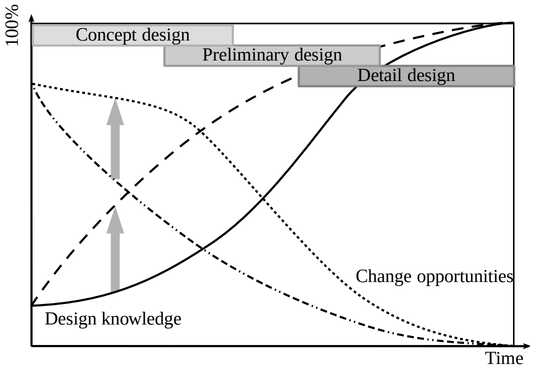

Fig. 1. KBE goals in concurrent development cycle, based on [9]

Generative Engineering Design is a process in which the draft or model is made based on quantitative and qualitative parameters or aesthetic inputs, using algorithms created by individual human intervention for the purpose of generating a variable set of subsequent models following accurate relations and according to hierarchical connections.

Even though similar, the fundamentals of Generative Engineering Design are different from parametric modelling. Generative engineering relies partly on generic parametric models. Parametric models are numerically controlled deterministic representations of design solutions which result in a new product with similar geometrical values (quantity indicators such as dimensions, weight etc.), but dissimilar in quality (e.g. aesthetic indicators, subjective user requirements, and needs). It means that generative design in new design and innovation offers more than a geometric model. It offers a whole complex of information about a new product which has not only a deterministic nature, but also a heuristic one.

Today, parametric modelling is a well established approach. While lower levels of parametrization (i.e. parameters and formulas) are commonly used, higher levels are still underutilized. These higher levels are made of composite geometry, which is required to dynamically adjust to variable inputs. Complex geometry brings more constraints that have to be met to produce a valid result. The parametrization levels, and some practical solutions for its high-level use are well illustrated in [3,4].

Knowledge-Based Engineering describes a wide range of CAD and CAE applications that are used in cooperation with each other, with emphasis on knowledge reusability in repetitive tasks following normal engineering practices [5]. Part of such a cooperation is related to the definition of specific parameters and other kinds of requirements on behalf of automatic design creation. GED is generally a part of such a methodology [6–8]. The proposed methodology addresses two of current shortcomings of KBE as identified in [9]. It tries to avoid the resulting model to be a 'black box' by providing explanations, links and context to data and formulas. The knowledge reuse problem is partly addressed by implementation in a well-established CAD software. The reliance on proven software is suitable for long-term usability or work in diverse environment, although there are limitations (lack of knowledge sharing between applications). Another problem with knowledge reuse in KBE application is the form, which usually prevents the reuse for non-programmers [10]. This is addressed by the chosen approach by relying mostly on visual and graphical workflow.

An overview of selected KBE goals is shown in fig. 1. To further leverage the time-saving and solution-finding advantages of concurrent product development cycle, it is needed to improve the speed and accuracy of information exchange. The shortened development time is attributed to more automation possibilities of repetitive tasks [9,11] and reusability of design elements. This has an impact on earlier stages, where the effect leads to better product knowledge. The improved design knowledge leads to more change opportunities, allowing wider exploration of design space [12]. In this paper, the focus is on the geometrical definition and the reusability of geometrical features.

2.1. GED methodology for development of shaped components

The unique advantage within the shape component design lies in the possibility to define models by the shape and position of surfaces. For example surface of contact is difficult to be set in solid modelling. The presented methodology is based on a proper hierarchy of operations throughout the shaped (i.e. surface-based) component modelling. Such a component is built step by step as is shown in fig. 2. The main advantage lies in the hierarchy of separate files created in CAD software:

Class A surface represents input surface. Quality of connections between patches, aesthetics, aerodynamics and passive safety are considered here.

Class B surface represents derived surface (offset from A-surface with added technological adjustments). It is important to link all the operations in a way that allows its adaptation to any changes of the class A surface.

Interface represents connection between parts. For each part pair a separate file is used, referenced by both parts (generally, an interface can be defined between more than two parts). In each interface file, separate geometrical set exits for particular interfaces (e.g. shape boundary, contact surface, clip definition). It is responsible for additional changes that can be made between parts, propagating the change to all interfaced parts. Interface creation is the best benefit of automatic modification of assembly during the subsequent stages of product development process. It means that in the case of a class A surface restyling, any affected parts would be changed automatically. This can be called a generatively transformed design.

Result represents the final result, solid or surface for analysis created from hierarchy of input parts and handling interface realization (e.g. shape trim, clip surface instantiation).

Functional features (class C surface) represents geometry creation which leads to a design of functional features such as clips, ribs, holes etc. A class C surface refers to a functional surface, which originates between parts as interface and after assembling it is constantly covered (not visible from any side). Class C surface is the main construction element of functional features.

2.2. Application of GED methodology during the virtual sport vehicle development

The proposed scheme shown in fig. 3 is suitable for designing process of vehicle body components. Development of virtual sport vehicle illustrates such a process. It is one of objectives in the Laboratory of Generative Engineering Design at the Slovak University of Technology in Bratislava. Firstly, a CAS