

Engineering Optimisation by Means of Knowledge Sharing and Reuse

Olivier Kuhn^{1,2,3}, Harald Liese,¹ and Josip Stjepandic¹

¹ PROSTEP AG, 11 Dolivostraße Darmstadt, Germany

{Olivier.Kuhn,Harald.Liese,Josip.Stjepandic}@PROSTEP.com

² LIRIS, University Lyon1, Bât Nautibus, 8 Bd Niels Bohr Villeurbanne, France

³ LSIT, University Louis Pasteur, Pôle API, Bd Sébastien Brant Illkirch, France

Abstract: Due to the increasing complexity of technical products and the decrease in product development time (“time to market”) the urgent need for the optimization of cross-skill engineering collaboration demands short-term solutions. In this paper we will therefore introduce an approach to federate the established engineering tools and processes on the basis of commonly shared engineering models rather than to integrate or to harmonize them in a long-term and highly complex development process. The underlying methodology for this service oriented knowledge sharing concept will be based on a semantically enriched, formal notation for the definition of the necessary data mapping & linking between the different engineering models. Besides an introduction to this new collaboration approach this presentation will provide an overview of the state-of-the-art CA technology and artificial intelligence technologies capable to contribute to the conceptual design of a short-term solution and point out the individual shortcomings, that are currently addressed by an ongoing project at PROSTEP in collaboration with the Université Claude Bernard Lyon and the Université Louis Pasteur Strasbourg (France). The paper includes examples with the first implementation of free form deformation algorithms, artificial neuronal networks and functional DMU.

Keywords: Design Optimization, KBE, Conceptual Design, Multidisciplinary Simulation, Mechatronics

1. Introduction

The hard competition in today's markets obliges companies to reduce costs and time to market while pushing ahead with innovation. Customers demand high quality products at low prices. Therefore, the development process has to be optimized regarding not only the tools being used, but also the way of working.

The integration of analysis modules (workbenches) in most design systems, and therefore the emergence of simulation-driven design, has characterised the last decades.

One way to improve development process is collaborative engineering. Today several competence fields are involved in design phases and experts from these domains have to work together. This collaboration involves data exchanges, knowledge sharing and knowledge reuse. To tackle this problematic we are collaborating with LIRIS¹ laboratory in Lyon, where a team is specialised in collaborative systems and LSIT² laboratory in Strasbourg where researches on data mining and evolutionary algorithms are performed, in order to model and optimise the collaboration between experts in design phases.

Heterogeneity of data and tools is a big problem in collaboration. That can lead up to errors and is time consuming. A solution to prevent misunderstanding is to specify formally the domain where experts are working and its semantic. This can be done thanks to ontology. Ontologies are used to define concepts and relations in a domain, in our case a domain related to experts environment. They also allow reasoning among concepts to discover new relations that are not explicitly defined.

Modelling and specification of collaboration and of the domain can bring huge time benefits because information will be understandable and reachable to each participant and also reusable between experts and projects.

To push further the optimisation of collaborative work, we resort to artificial evolution. At several level in collaboration, things can be represented under graph form, for example exchanges between participants in collaborative design and for each level different objectives can be defined. A method to enhance current state of graphs is to use evolutionary algorithms applied to graphs to get nearer from objectives.

The merge of semantically defined domain with ontology and evolutionary algorithms is an interesting challenge which will certainly lead to an enhancement of cross-skill collaboration and thus reduce time to market and costs (see Fig. 1).

The most used approach is CAD centric. Generally, the basic CAD systems have been enhanced with Simulation modules (FEM, Multi Body Simulations etc), thereby enabling designers to perform first analyses before finishing the design models.

However, further simulation types are not yet implemented. According to most processes, CAD models are reviewed by senior designers and afterwards confirmed by the simulation department. Thus simulation engineers run analyses and make some change requests if necessary. After processing the change requests, designers have to improve the models and run further reviews, until the components are confirmed by the simulation department. Depending on some

¹ <http://liris.cnrs.fr/>

² <http://lsiit.u-strasbg.fr/>

factors, such as the complexity of the product to be developed, the size of the company as well as its internal organisation, many loops can be run between designers and simulation engineers.

This leads to a long product development time. Therefore, reducing the loops between these departments, through the use of analysis tools embedded in CAD tools, contributes to shortening the time to market.

These analysis tools may be used to perform first calculations and in certain cases some standard simulations, which might have been defined by the simulation department in advance.

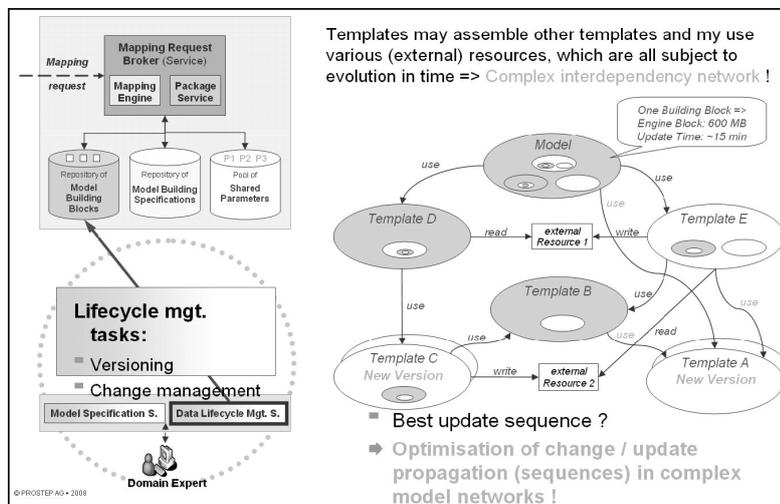


Figure 1 Model building and data mapping

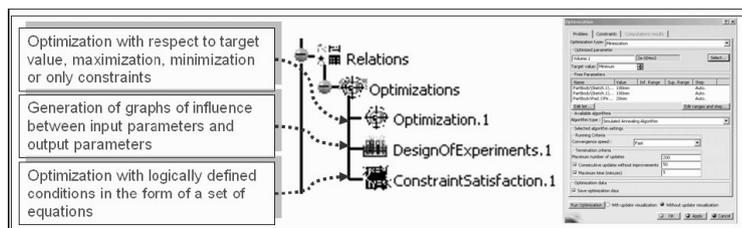


Figure 2 Design Optimization with CATIA V5

2 CAD Internal Analysis Tools

Examples of analysis and optimization tools embedded in design systems are the Product Engineering Optimizer workbench (Fig. 2) and the FEM-Analysis workbench of CATIA V5.

Additionally, some KBE features (Knowledge Based Engineering) can be used to perform other analyses such as automatic variation of design parameters.

Furthermore, simulation-driven design may rely on templates and processes, which might have been defined by the simulation department in advance. In these cases, designers run predefined processes in order to increase effectiveness.

However, the tools that are integrated in CAD systems are not appropriate to solve some specific problems, such as surface optimization, which plays an important role in disciplines such as aerodynamics.

In addition, these embedded tools generally have been developed in order to be run on a single machine. Therefore, it is difficult to perform some analysis without facing performance problems. This is especially the case when a designer attempts to generate a big number of variants and to evaluate them in a short time. Furthermore, the full potential of KBE techniques is still not used by these tools, although they can support designers for tasks such as variant evaluation.

3 Surface optimizer for Geometric Models

The optimization of technical components and products has gained significance in the last decades and belongs nowadays to the classical activities of product development.

Some application areas are the optimization of aircrafts wings in order to influence the lift, the volume optimization in the mechanical engineering and the area optimization in the automobile industry. Optimizations may be performed according to topology, form, dimensioning and material. The objective of form optimization is the deformation of a model's area, without topology modification [1].

3.1 Requirements for the Surface Optimizer

Among other requirements at the beginning of the development of the Surface Optimizer, the quality of results and computation time were very important. In addition, its integration into the CAD environment with which designers were familiar, had a high priority. Given the fact that CAD models may be arbitrarily complex, it was necessary to use an optimization method that was independent, not only from the parameterization of the geometric model, but also from the CAD system. The concepts had to be implemented in CATIA V5.

3.2 Concept

The high complexity of CAD models runs hand in hand with a big number of optimization parameters. Therefore, the analysis and evaluation of the model to be optimized can be a tedious task, because the number of model parameters influences the computation time. Free-form deformation (FFD) is suitable to

tackle these types of challenges, because instead of the CAD model, its surrounding space is deformed [2]. Consequently, the optimization refers to the parameters of the surrounding space.

However, free-form deformation is not an optimization method. Therefore, algorithms are needed to compute the search direction of the target value and the increment. After the evaluation of some optimization methods, the Quasi-Newton method was selected because of its integration characteristics and performance.

The objective of the optimization was the minimisation of the CAD model surface. The CAD model is then updated and visualized by users. For this purpose, special functions are used by free-form deformation to move the vertices of the grid surrounding the model [3].

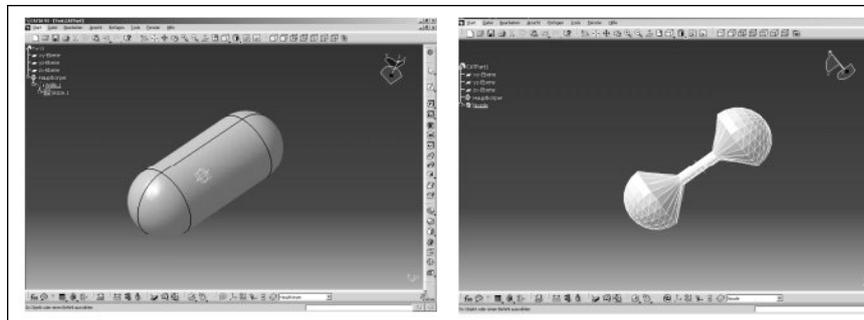


Figure 3 Generation of an object with minimal surface area according to constraints

3.3 Results and Integration into CATIA V5

The CATIA API (CAA) was used to integrate the surface optimizer into CATIA V5. Concerning the users interaction, the scenario consists of selecting a CAD model in the CATIA environment, fixing the areas that should be invariant during the optimization and launching the analysis. Fig. 3 illustrates an example of a simulation for which two areas (the ends) have been defined as invariant.

This solution differs from similar applications in that the optimization is not a pure mathematical variation of parameters. Instead, designers can define constraints with the interaction techniques they are used to. This is realised through the selection of areas of the model defined as invariant. Designers are therefore freed from the task of thinking about the mathematical foundations of the program before performing optimizations.

Apart from surface and volume optimization, a further application area is the packaging or enveloping of mechanical components. The principles of the optimizer may be applied to quickly derive appropriate envelopes from mechanical components or systems such as engines.

4 EvoCAT: Using Artificial Intelligence to Accelerate Conceptual Design

The phases of product design are planning, conceptualization, embodiment design (geometric modelling) and elaboration (detail design) [4]. Processes for the embodiment design as well as for the product elaboration phases are supported by diverse computer aided tools. However, product planning and conceptual design are still not efficiently supported.

Although diverse CAD systems have integrated optimization modules, designers still have difficulty influencing the results, especially from a design point of view.

The results that are delivered, for instance in the case of CATIA V5, are summarized in a table, whereby an end solution is presented to the user. Therefore designers have to analyse these values in order to evaluate different configurations. This task is very tedious when a large number of variants is available.

The challenge in this project consisted not only in generating variants, but also in evaluating these variants and selecting a small sample of them. This should be managed by designers. For this purpose, knowledge-based methods have been implemented in order to perform some decisions, but also in order to reduce the computation time.

EvoCAT is a system that enables designers to optimize models according to specific criteria. It makes use of evolutionary algorithms in order to support designers with optimizations in early stages of product development [5]. The capacity of evolutionary algorithms used in this case is the ability to choose an almost optimal variant from an arbitrary start population. Thus, these features of EvoCAT are suitable for generating and evaluating concepts. However, a productive use necessitated the involvement of further artificial intelligence methods.

The optimizer (EvoCAT) interacts with many CAD sessions (CATIA V5 in our case, but other systems may be used), which perform computations. Due to the large number of computations that are performed, the time that is necessary for completing optimizations is not satisfying without further adaption. Accordingly, neural networks have been implemented with the objective of taking over the job of the CAD system after a training phase. Practically, the neural network that has been implemented approximates the computations of the CAD system and enables, thereby, a reduction of the running time of EvoCAT.

4.1 Requirements

In fact, the aim of the main program consists in generating design variants and evaluating them in order to free designers from that task. This would be

impossible to fulfil without computer support, in view of the large number of variants that are generated. Furthermore, the usability of the program is decisive.

A further requirement consists in enabling designers to use the optimization system without needing the help of a software specialist to integrate certain algorithms into the CAD system. Furthermore, designers have to be exempt from the task of importing computed variants into the CAD system. In fact, many optimizers deliver results outside of the CAD systems that contain the model. Therefore data inconsistencies may arise when switching from one system to another.

One of the well-known challenges of automatic variant generation is their evaluation and classification, because very large numbers of variants may be created. A solution had to be found, that enabled the classification of generated variants and therefore provided designers with information that facilitated decision-making. Accordingly, designers were to be exempt from checking many thousands of variants.

4.2 Concept

The main building blocks that are used for generating and evaluating variants are EvoCAT, the CAD system and an artificial neural network. In order to obtain a high level of performance, the calculations are performed in a distributed environment.

The technical approach of EvoCAT relies on using characteristics of evolutionary algorithms. The latter emulate the biological evolution by simulating phenomena such as reproduction, mutation, selection and survival of the fittest. Moreover, the approach makes use of probabilistic rules. These principles are applied in order to evaluate the variants generated by the CAD system in respect of criteria that have been defined by designer EvoCAT is made up of two main components, the server and the client. The server addresses the CAD system and the artificial neural network as well. Results are sent back to the client that enables visualization.

The geometric model is the basic element of the concept:

- Input and visualization are taken over by the CAD system
- The CAD system is used as a computation and simulation tool that generates variants
- Data is available in diverse formats (CAD, CAE, export formats)

The third main component of the optimization system is the artificial neural network (ANN) that is involved in order to reduce computation time.

The ANN is trained during optimization and evaluated after each training session. Once the quality of the ANN has reached a satisfying level, it is involved in the calculation of target functions. In fact, the ANN takes the job of the CAD system by approximating its values; therefore EvoCAT obtains return values quicker than if the CAD system would have computed them. Doing so shortens

the running time of EvoCAT and consequently the reaction time of the system. However, the quality of the results obtained remains decisive. Thus if the ANN delivers incorrect results, it is set once more in training mode. For this purpose, a monitoring approach has been realised.

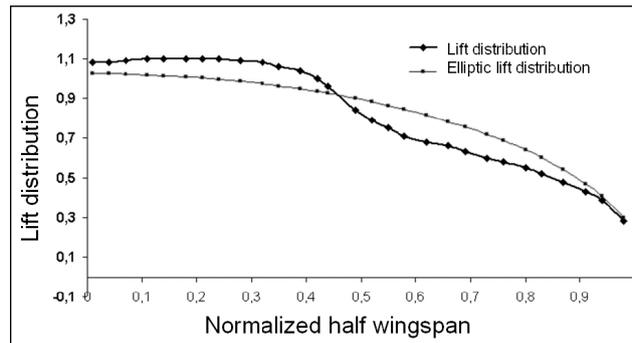


Figure 4 Lift distribution of a flying wing by extended flaps

4.3 Results and Integration into CATIA

EvoCAT has been applied for dimensioning and optimizing a flying wing. For this study, an additional module was involved for calculating aerodynamic coefficients.

The extended flaps increase the lift, however they change its distribution and therefore worsen the induced drag (Fig. 4). In fact, the elliptical curve is ideal for induced drag distribution.

The task of EvoCAT consisted in finding a wing form that provided an elliptical lift distribution even if the flaps were extended.

For optimization purposes, a so-called K-factor [6] had to be defined. It designated the ratio of the induced drag coefficient over the induced drag coefficient of an elliptic lift distribution. In addition, the flight stability was checked too. In so doing, it could be insured that the flying wing was not only theoretically dimensioned, but also could fly stably.

The flight stability factor and the K-factor were combined to determine a specific factor that was to be evaluated by EvoCAT.

As a result, an improved wing geometry that delivered best stability and lift distribution was expected. For test purposes, a student who was a non-expert in aerodynamics used EvoCAT, while an experienced aerodynamic engineer analysed the problem. Both test persons obtained the same results. However, the student was quicker.

The best variants that were computed by EvoCAT were visualized in CATIA V5 using design tables (Fig. 5). Real tests have confirmed the results obtained.

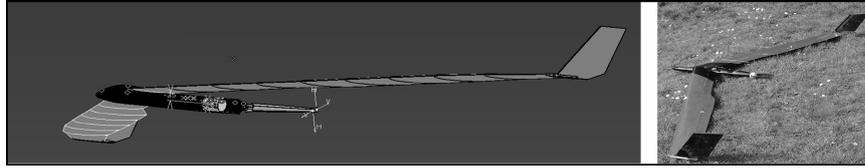


Figure 5 Carbon fibre reinforced flying wing with 2,6m wingspan

5 Functional DMU: Adjustment of Mechanical Components with Control Systems

Whereas simulation fields are traditionally quite disconnected from design (heterogeneous application landscape), one of the main choices for companies is to adopt a multi-disciplinary approach to attain a global optimization of their products. This is particularly appropriate for mechatronics, where mechanics, electronics and software are involved in the development of the final product.

During the design of a mechatronics product, it is very important to manage the interaction between the different disciplines, in order to develop suitable solutions.

According to an Aberdeen Groups benchmark report [7], the challenges for product development in the mechatronics domain rely on the link between disciplines.

That study illustrates that three of the top five challenges of mechatronics product development are related to a lack of integration in the development process.

5.1 Requirements

Different levels of coupling CAD software and behaviour simulation software have been proposed in the past, from the coupling of information in each component to the development of environments that integrate all the simulation domains [8, 9].

One of the main approaches currently used consists of extracting information from mechanical design data and inserting it into the simulation model. Traditionally, the transfer of information has been done manually, but automatic or semiautomatic approaches have been developed to map behaviour models onto mechanical models [10].

Although simulation tools enable the modelling of product behaviour, its simulation and analysis of its results, they are not focused on managing geometric information and its visualization in 3D environments. Therefore either the visualisation of the product is not possible with these approaches, or 3D models are reduced to simplified analogous models (Fig. 6). The perception of designers is consequently limited.

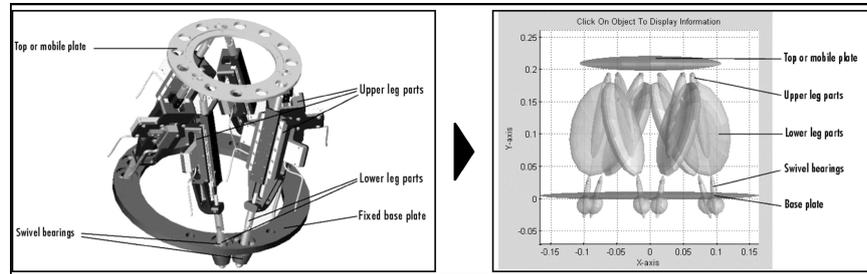


Figure 6 Simulation and visualization of a Stewart platform with SimMechanics[10]

Considering the arguments mentioned above, an interface that takes advantage of both CAD and control system design software was to be developed. In addition, the system had to enable simulation of the behaviour of a complex product from the early stages of the development.

To ensure the numeric coherence, it was important to synchronize the execution of both applications (CAD and behaviour simulation). Indeed, the values that are used for computation in the simulation tool must continually correspond with the positions and properties of the parts in the geometric model.

Furthermore, the realistic visualization of the simulation was mandatory. Modifications during multi-disciplinary simulation had to be considered immediately by the simulation loop in order to avoid data inconsistencies.

5.2 Concept

The CADSim Interface is independent of the software used for validation (CATIA and MATLAB). In practice, the CADSim Interface has been designed to be easily adaptable to other software. Therefore the tools that have been linked at this time can easily be replaced by a different CAD and simulation system.

Furthermore, the creation of a concept for coupling mechanical and behaviour models involves questions about the modelling of control system components and the communication of both models.

Starting from the observation that in the physical world, there are elements necessary to link the control system with the mechanical components, the key elements of the CADSim Interface are sensors and actuators. They are independent from both CAD and simulation-based applications, and they are linked with modules that manage the communication with CATIA V5, on one side, and with MATLAB on the other (Fig. 7).

5.3 Results and Integration into CATIA V5

The CADSim Interface has been developed in a project as an add-on for CATIA V5, and it uses some methods provided by MATLAB to realise the

communication between both applications (COM approach), which are running simultaneously.

Changes performed on the models are immediately considered by the simulation loop.

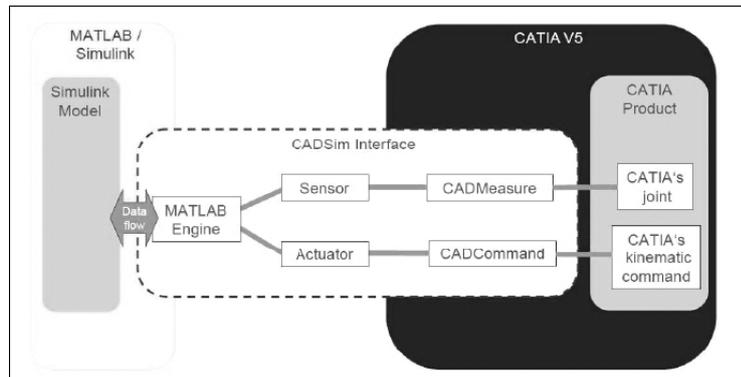


Figure 7 Concept of CADSim Interface

In order to run the simulation, an arbitrary CATIA model is to be loaded. After doing that, the user has to select the commands that should be considered as actuators and the components on which sensors values are to be measured. The CATIA specification tree is used for performing that selection.

In order to complete the configuration, the user indicates an arbitrary MATLAB model that should be loaded to run the simulation. The sensors and actuators of the geometric model are then mapped to corresponding objects of the selected MATLAB model.

The configuration file that contains this information is saved. Therefore the users are exempt from achieving configuration tasks as long as the same digital models are adjusted. Various tests have been performed with models of aircrafts and industry robots (Fig. 8).

Plots present the progression of current values and reference values during the simulation. Coupling that information with the realistic visualisation, designers and control system engineers can improve their digital models. In practice, control system engineers may define preliminary prototype models with minimal requirements (in fact features). Designers would already be able to adjust their CAD models to some preliminary behaviour models stored in a library during the first preliminary studies. That is the purpose of Functional DMU.

6 Summary

This paper has described projects of PROSTEP AG dealing with the reduction of product development time through accelerating the design process, stimulating

the creativity of designers and synchronizing the development of partial product models.

An optimizer for surface minimization as well as EvoCAT for rapid generation and evaluation of design variants has been described. In addition, an interface that enables the coupling of mechanics and control systems for the development of mechatronics products has been presented. The applications are being steadily customized in order to satisfy the specific requirements of different customers.

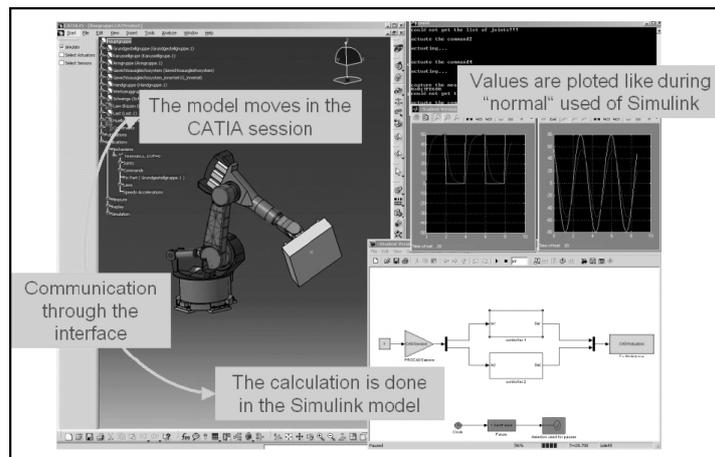


Figure 8 CAD model of industry robot being controlled by a MATLAB model

References

1. Kemmler R.: Stabilität und große Verschiebung in der Topologie und Formoptimierung, Dissertation, Institut für Baustatik der Universität Stuttgart (2004).
2. Sederberg T.W., Parry, S.R.: Free-Form Deformation of Solid Geometric Models. Proceedings of SIGGRAPH '86, Computer Graphics 20(1), pp. 51-159 (1986).
3. Hechler J.: Formoptimierung unter Nebenbedingungen durch Freiform-Deformation. Dissertation, Fachbereich Mathematik der Technischen Universität Darmstadt (2006).
4. Pahl G., Beitz W., Feldhusen J., Grote K.H.: Konstruktionslehre - Grundlagen erfolgreicher Produktentwicklung, 6. Auflage, Springer (2005).
5. Slaby E.: Einsatz Evolutionärer Algorithmen zur Optimierung im frühen Konstruktionsprozess. Dissertation, Universität der Bundeswehr Hamburg (2003).
6. Nickel K., Wohlfahrt M.: Schwanzlose Flugzeuge - Ihre Auslegung und ihre Eigenschaften, Birkhäuser Verlag (2008).
7. Jackson K.C.: The Mechatronics System Design Benchmark Report - Coordinating Engineering Disciplines, Aberdeen Group report (August 2006).
8. Sinha R., Paredis C., Khosla P.: Integration of mechanical CAD and behavioral modeling. Proceedings of the IEEE/ACM Workshop on Behavioral Modeling and Simulation (2000).
9. N.N.: SimEnterprise-Extending Simulation to the Enterprise. MSC White Paper (2006).
10. N.N.: SimMechanics 2 - Users Guide (2007).