CONCEPTUAL DESIGN OF HETEROGENEOUS SYSTEMS

Michael Reeßing, michael.reessing@tu-ilmenau.de
Ulf Döring, ulf.doering@tu-ilmenau.de
Torsten Brix, torsten.brix@tu-ilmenau.de
Technische Universität Ilmenau, Germany

Abstract. *In the past it was often adequate to assemble an overall system from separately developed and optimized parts. However, recent developments in engineering show the need to integrate mechanical, optical, electrical, electronical and software components. This new quality of interdisciplinary collaboration requires new computer-aided, phase- and domain-spanning tools for modeling, analysis, simulation and optimization of complex design objects particularly in the early phases of product development. The article covers the first intermediate results of the ongoing research concerning a phase-overlapping design system for heterogeneous systems that supports the transition from the solution principle to embodiment design.*

Keywords: *Heterogeneous Systems, Early Phases of Product Development, Constraint-based Modeling*

1. INTRODUCTION

For computer-aided modeling and calculation of heterogeneous systems different approaches are known. Some of them already allow an holistic development process for (usually mechatronic) components and products.

The following categories exist:

- general simulation systems,
- specialized simulation systems,
- parallel application of multiple simulation systems,
- guidance and assistance systems and
- integrated solutions.

For the design and particularly the simulation in early phases **general simulation systems** can be applied. These systems are flexible since modeling is done by entering source/script code, equations or schematic diagrams that describe and simulate heterogeneous systems on a functional level. However, this approach leads to a number of disadvantages. It is often difficult to derive equations that represent functional properties and to incorporate structural properties (e.g. function-relevant positional relations of mechanical and optical components). Also, there is usually no support for the transition from the functional description to the embodiment.

Specialized simulation systems are usually not suitable for holistic modeling and simulation of heterogeneous systems. For instance, the modeling system for electronic circuits SPICE allows limited handling of non-electronic components that have to be transformed into equivalent electronic or electric circuits. It is not possible to model components for feedback control systems (Roddeck, 1997). Serious limitations concerning a uniform equivalent description of heterogeneous systems also exist in dynamic simulation systems (e.g. ADAMS, ALASKA, SIMPACK, ITI-Sim), in software for control theory (e.g. FSIMUL, SIMULINK) or in optics simulation systems (e.g. OSLO, ZEMAX, DIFFRACT). None of these systems closes the gap between the solution principle and the embodiment with focus on the transition between the two representations.

The parallel application of multiple simulation systems considers the necessary multidisciplinarity of the design of heterogeneous systems as the user applies a number of different, domain specific softwares that support modeling, analysis/simulation and evaluation/optimization. This approach demands a high degree of design experience and the mastery of many software products. Also, redundancies and loss of information are inevitable in such a process. To avoid the resulting additional expenses, it is reasonable to join the different softwares that cover only narrow fields of representation, calculation, simulation and optimization.

One possible approach for this are **guidance and assistance systems (**Pahl and Beitz, 2003). These provide a methodical guide with specifications for merging the data of a design object in a uniform repository of data. The core of such systems are usually CAD systems that are linked with other softwares to support the design of certain products or product groups.

Also, modern CAD systems are able to provide **integrated solutions** for several problems concerning the design of heterogeneous products. Typically, this is done by adding customized add-on modules for certain scopes of problems offered by various vendors. Examples are add-ons for multi-body simulation (e.g. Autodesk Inventor and ALASKA), for the finite element method (e.g. SolidWorks and COSMOS) or for design of drives and controllers. However, there is no CAD system that covers the whole spectrum of requirements for the design of heterogeneous systems.

In conclusion, the mentioned approaches show deficiencies concerning the domain-overlapping description of heterogeneous systems and the phase-overlapping modeling from the functional description to the embodiment design.

2. CONSTRAINT BASED MODELING

Models of heterogeneous systems can be seen as a set of components and a set of relationships between these components. In the following text these relationships are referred to as constraints.

Usually Constraints are defined between the parameters which describe the properties of the components (shape, position, material, physical and electrical properties, etc.). If the parameters have values such that all constraints are fulfilled, the model is called to be consistent.

After changes of some parameters (e.g. during an optimization) or even often in the initial state the constraints are not fulfilled. Then calculations have to be performed to find new values of dependent parameters which make the model consistent (Fig. 1). If not all constraints can be satisfied different strategies may be applied, e.g. restoring the last consistent state, minimization of the errors (optimization) or ignoring certain constraints (that means the model could "break").

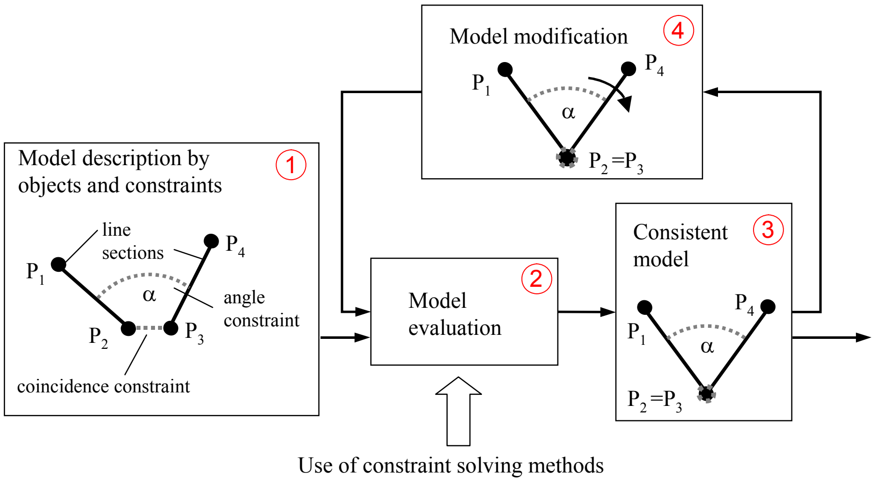


Figure 1. Calculation cycle in constraint based modeling.
(1) Initial model definition (two line segments, coincidence constraint between P_2 and P_3, angle between lines).
(2) Model evaluation that leads to (3) a consistent state (all constraints satisfied).
(4) Any modifications to the model (e.g. by user interaction) require a re-evaluation of the model.

In addition to constraints which reference values of parameters (e.g. those parameters describing more complex elements like cylinders or rays) some types of constraints control the existence of such elements or components. This includes the existence of the single parameters as well as according constraints referencing the parameter values.

In general constraints can be defined by modelers in very different ways (Brix, 2001), e.g.:

- equations and inequalities (very general formulations),
- clauses in predicate logic (e.g. to handle existence),
- fuzzy relations (to represent fuzzy information) and
- domain specific relations (often those relations cover sets of values, e.g. position parameters in assembly constraints).

Domain specific relations like assembly constraints may be formulated as equations and inequalities too. During such a transformation domain specific information is lost, thus it can not be used in the calculation process. If the domain specific information about the semantics of relations is kept then appropriate calculation modules can be chosen and the handling of such relations becomes much more efficient (example: iterative solution of mostly non-linear equations using Newton-Raphson method vs. geometric constructions using a ruler and compass approach). Furthermore special cases (e.g. degenerated cases, multiple solutions) can be handled in a way which is expected by the user. For interactive work both advantages of using domain specific information are important. The engineer should realize the consequences of his design decisions (e.g. chosen parameter values and according product properties) as fast as possible and he should never be irritated by the presentation of solutions which are formally correct but totally unexpected. A very simple example is shown in Fig. 2.

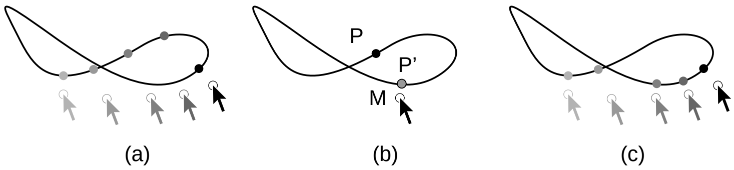


Figure 2. Behavior of a point on a path of movement.
(a) A point on the path is moved depending on the position of the mouse cursor. As the user would expect, the point stays close the curser and moves continuously on the path.
(b) P and P' are possible positions for the mouse cursor input M.
(c) Sequence of point positions that is not expected by the user (not continuous according to the path).

This shows that the heterogeneous formulation of heterogeneous systems is reasonable according to calculation efficiency as well as solution plausibility.

The choice how relationships are to be expressed should be made by the engineer who builds up a certain model. Nevertheless internally in the constraint solving application the model representation may be transformed with respect to available calculation modules. Quite simple is for instance the transformation of geometric constraints into equations. The inverted direction (extraction of geometric constraints from equation systems) is more complicated in general. As an example a distance constraint between two points may be used. For its detection the variables storing the point coordinates must be identified (the number of coordinates depends on the dimension) and a further parameter must be identified as the distance. The quadratic equation which references these parameters is a strong hint that the engineer had the intension to model a distance relation but in general that is not sure.

3. INTEGRATION OF DOMAIN-SPECIFIC CALCULATION MODULES

One of the prerequisites for a design system capable of integrated modeling and simulation of multiple domains is the representation of the model in one single graph or network. This model describes components from all involved domains including their properties and relations in a uniform way. Usually, there is no calculation approach to handle such a representation as a whole. Therefore, parts of the graph need to be treated by specialized calculation modules (Fig. 3).

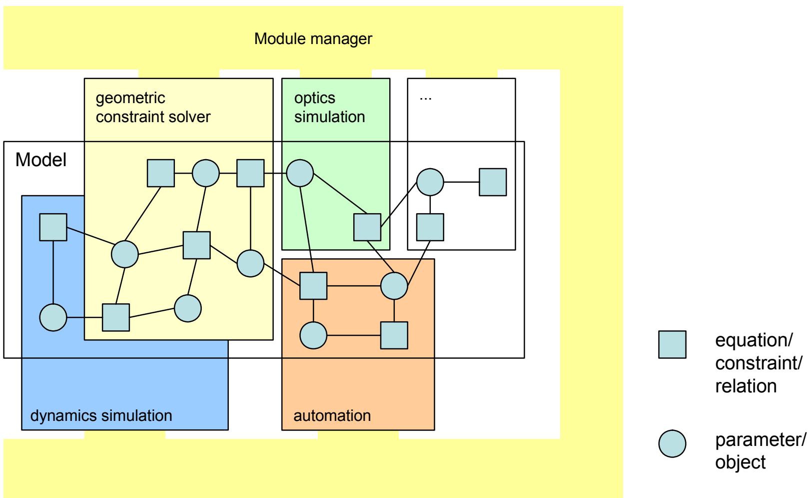


Figure 3. Specialized calculation modules/solvers handle parts of the model. In some cases a model part can be calculated by more than one module. Then, the module manager decides which module handles this part depending on the task.

A main problem concerning the calculation to achieve model consistency is the choice and coordination of the calculation modules. In the phase-overlapping design system MASP (Modeling and Analysis of Solution Principles), this is done by a module manager. The implementation of this coordinator can be based on one of the following options:

- Transformation of all constraints into equations and inequations (differential equations if necessary) and algebraic (only for small systems of equations) or numerical (initial approximation problem) solution.

- All dependencies are compatible to a single calculation module, i.e. such a module can handle them directly or a transformed version of them. In this case that module alone can generate a consistent model state.
- The model is divided into parts that can be handled by a set of specialized calculation modules.

The latter approach is not only the most general and efficient one, it is also more difficult to implement. Amongst others, the following problems need to be solved:

- Segmentation of the model parts to be solved

One aspect is the intention to solve constraints at a high level of abstraction to benefit from the advantages of domain-specific solution methods. On the other hand, in order to limit the complexity of the coordination of the solver modules a few large model segments are easier to handle than many small ones. Therefore it makes sense to support some general solution tools in the specialized calculation modules (e.g. support of simple equations).

Other aspects are the complexity and solvability of the model segments (Hoffmann et al., 2001a and 2001b). Particularly important is the avoidance of over-determination.

- Coordination of the calculation modules

For the calculation methods in each module exist input and output parameters. Input parameters can be constants, output values of other modules or user inputs. The module manager needs to coordinate the modules in a way that ensures a global solution progress, i.e. that calculated parameters of one model segment serve as input values of another segment (sequential approach). Since this is not always possible, an alternating or iterative application of solvers can be necessary.

Figure 4 shows an example for the alternate application of calculation modules. The symbolic representation of the model (slider crank, upper left corner) is a very abstract, domain-specific description (assembly of mechanical elements).

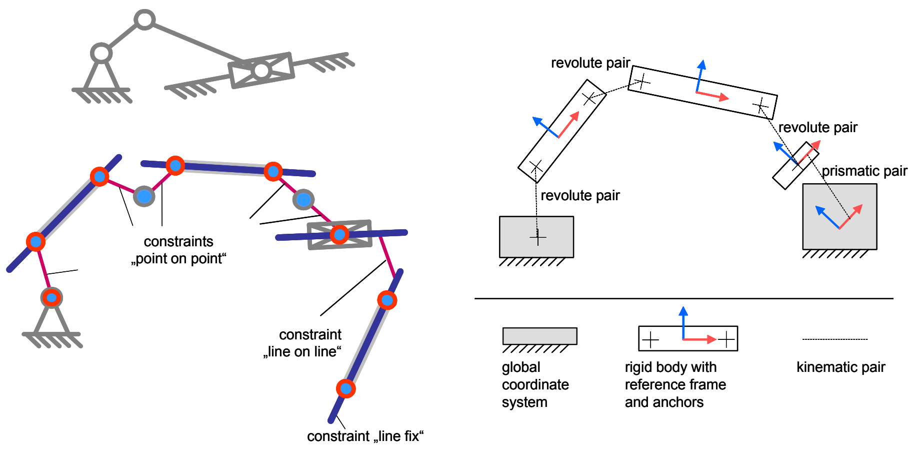


Figure 4. Transformation of the abstract model description (upper left) of a slider crank into specific representation for a geometric constraint solver (left) and for multi-body simulation (right)

For calculations the symbolic representation needs to be transformed into a solver-specific description. A conversion for a geometric constraint solver is composed of basic geometric elements and constraints (left side). On the other hand, a conversion for a multi-body simulation is described by rigid bodies and kinematic pairs (right side). Depending on the task the used representation (and with it the calculation module) is selected. While the geometric constraint description is favorable for an interactive modeling process, the multi-body representation allows calculations concerning the dynamics of the model.

The described constraint-based modeling approach allows a phase-overlapping description of heterogeneous systems since the generic model representation is independent from visualization and calculation (Brix et al., 2003).

4. APPLICATION EXAMPLE

As an example for the modeling of a heterogeneous system a micrometer is used. The micrometer is based on a revolvable/pivoted parallel glass plate. This kind of micrometer can be used as an adapter for measuring telescopes. The idea is the generation of a parallel offset from the principal axis by tilting the glass plate (Fig. 5).

With this parallel offset, deviations of objects can be measured in the range of micrometers e.g. by using a knurl with a linear scale.

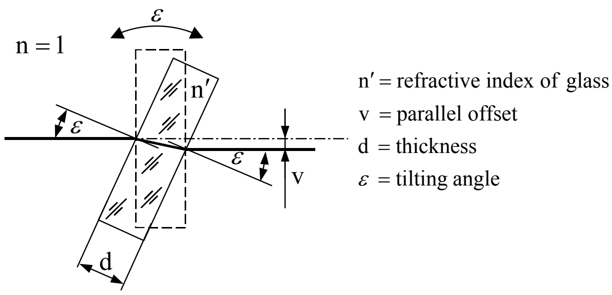


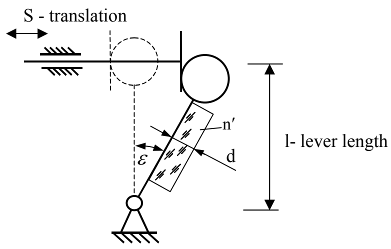
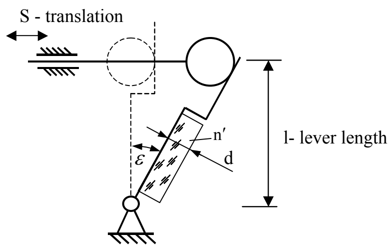
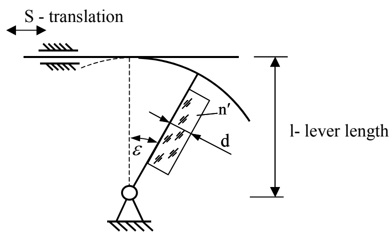
Figure 5. Parallel offset of the main beam by a coplanar glass plate

Starting point of the micrometer design is the identification of the functional interrelations between the tilting angle ε of the glass plate and the parallel offset v of the main beam. For this identification the application of the law of refraction is necessary. After some transformations the following equation results:

$$\text{v} = \text{d}\ sin\varepsilon \left(1 - \frac{cos\,\varepsilon}{\sqrt{\text{n}'^2 - sin^2\,\varepsilon}}\right) \tag{1}$$

To allow the measuring scale to be linear, it is necessary to find suitable function elements for an adjustment mechanism that linearizes this non-linear relation. For this, a simplification of equation (1) is convenient in order to decrease the design effort. Possible simplifications are the approximations shown in Table 1. These approximations can be found by assuming the tilting angle ε to be small.

Table 1. Approximations of the transfer function (1) to achieve linear behavior between input parameter s and tilting angle ε

Approximation	Solution principle
$v \approx v_{approx\_sin} = d\ \sin\varepsilon \left(\frac{n'-1}{n'}\right) = d\ \frac{s_1}{l}\left(\frac{n'-1}{n'}\right)$	 sine mechanism
$v \approx v_{approx\_tan} = d\ \tan\varepsilon \left(\frac{n'-1}{n'}\right) = d\ \frac{s_2}{l}\left(\frac{n'-1}{n'}\right)$	 tangent mechanism
$v \approx v_{approx\_\varepsilon} = d\ \varepsilon \left(\frac{n'-1}{n'}\right) = d\ \frac{s_3}{l}\left(\frac{n'-1}{n'}\right)$	 ε-mechanism

For the approximations in the first column of Table 1 simple adjustment mechanisms as solution principles can be found (second column in Table 1). These principles can be modeled in the parametric design system MASP (Modeling and Analysis of Solution Principles). Thus the determination and verification of function relevant properties are possible.

To evaluate the suitability of the different approximations (Table 1) a comparison with the original function (Eq. 1) is necessary. The calculation of the difference Δv between the parallel offset v_{approx} of the approximations and the parallel offset v of the original function (Eq. 1) can be done using motion simulations (automatic or interactive) of the full-parametric model in the design system MASP (Fig. 6).

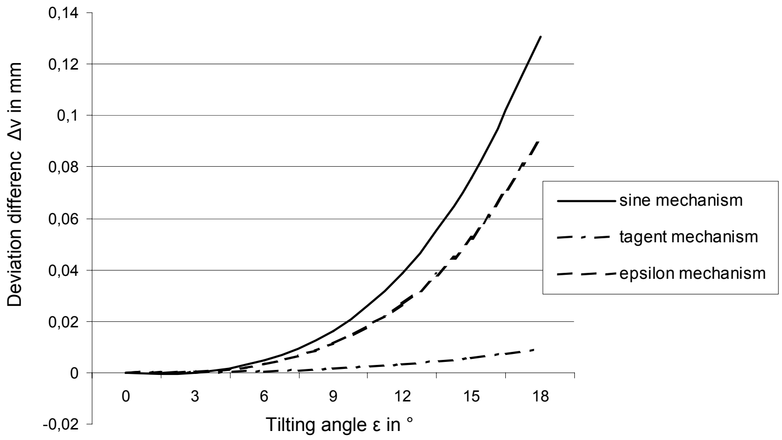


Figure 6. Difference Δv between vapprox of the approximations from Table 1 and v of the original transfer function (Eq. 1) for different tilting angles ε (assumption: refraction index n=1,519; thickness of the glass plate 22,44mm).

As the next step in the design process the adjustment mechanism with the smallest difference Δv to the original function is choosen to maximize the tolerance for errors caused by manufacturing and usage. The selected mechanism must be adapted to other conditions like installation position or sensitiveness of the adjustment mechanism including the knurl (Fig. 7). For this, a number of parameters must be specified with their interdependencies.

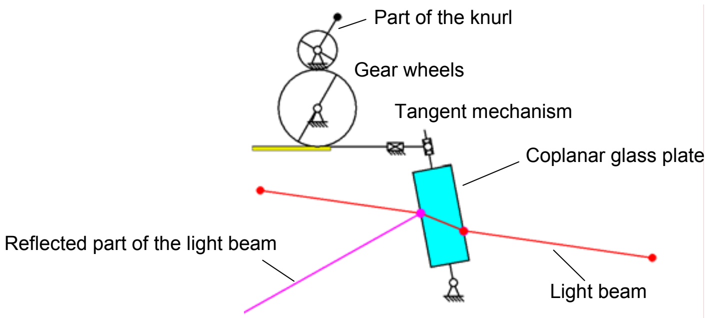


Figure 7. Solution principle of a micrometer modeled in the design system

Even this simple model allows many modifications by parameter variation (e.g. change of diameter of the gear wheels, thickness or the refraction index of the glass plate, parallel offset of the principal beam). The interactive graphical modeling of the solution principle is based on predefined symbols. The interdependencies between all parameters are handled automatically. Some examples of modification results are shown in Figure 8.

In this way a solution principle can be adapted to the required function and can be used to generate the first embodiment design.

To achieve an easy usability a catalog-oriented design is used in MASP. It supports the user in modeling the intended product attributes (e.g. layout/form, material, technological properties) of the first embodiment design using predefined solution elements in combination with invariant information in the solution principle like certain distances and angles or position of translation and rotation axes. Necessary bi-directional references between solution principle and embodiment design are added automatically during the generation. This allows iterations during the design process without losing the consistency between the model parts that describe the solution principle and the embodiment.

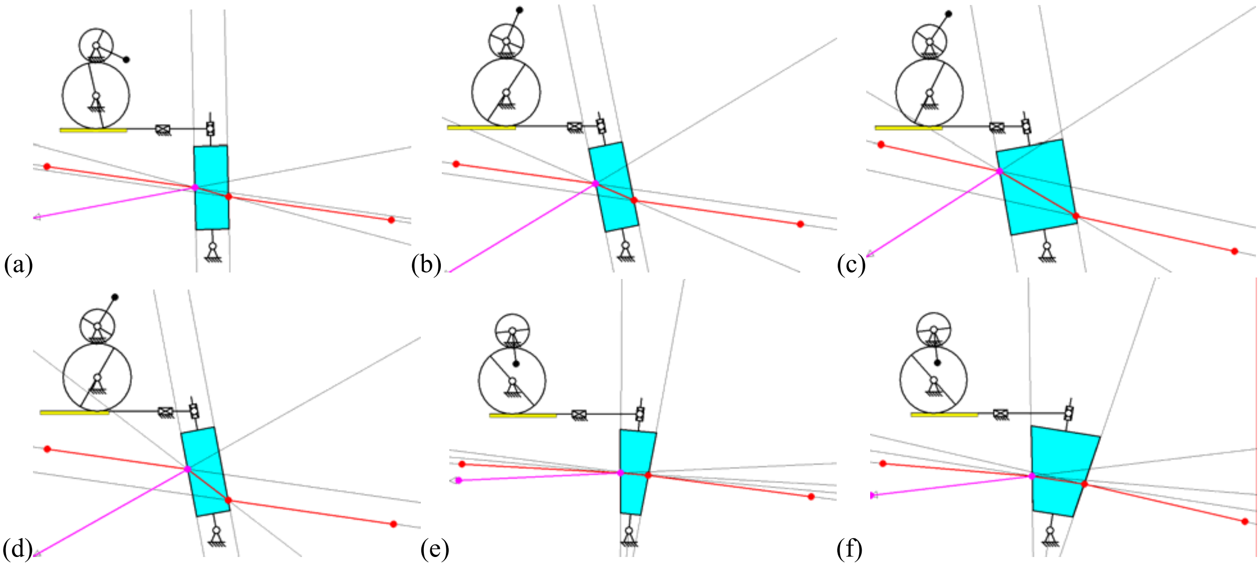


Figure 8. Modifications of the solution principle of a micrometer modeled in the design system MASP (the shown geometric sub-elements like directed lines illustrate the underlying constraint model). (a) and (b) Calculation of different positions of the micrometer by interactive rotation of the knurl. (c) The variation of the parallel offset with immovable adjustment mechanism and constant refraction index causes a change of the glass thickness. (d) The variation of the parallel offset with immovable adjustment mechanism and constant glass thickness causes a change of the refraction index. (e) The coplanar glass plate changed into an optical wedge. (f) The variation of the parallel offset with immovable adjustment mechanismus and constant refraction index causes a change of the wedge thickness

5. SUMMARY

The paper describes a concept as well as the first results of a project that aims to increase the effectivity of the embodiment design process for heterogeneous products. The concept integrates phase-overlapping as well as domain-overlapping design approaches into a computer-aided modeling system aimed at the early phases of product design. The basis for the presented work are solution principles that can be analyzed, simulated and optimized regarding different function-related aspects. The constraint-based modeling approach allows to include embodiment properties of heterogeneous models and to handle them in a holistic way. The optimized solution principles are the basis for the determination of an appropriate first embodiment of the function-relevant mechanical, optical and drive components.

6. REFERENCES

Pahl, G., Beitz, W., 2003, “Konstruktionslehre”, Springer, Berlin, Germany.

Hoffmann, C.M., Lomonosov, A. and Sitharam, M., 2001a, “Decomposition Plans for Geometric Constraint Problems, Part I: Performance Measures for CAD”, Journal of Symbolic Computation 31(4), pp. 367-408.

Hoffmann, C.M., Lomonosov, A., Sitharam, M., 2001b, “Decomposition Plans for Geometric Constraint Problems, Part II: New Algorithms”, Journal of Symbolic Computation 31(4), pp. 409-428.

Roddeck, W., 1997, “Einführung in die Mechatronik”, B.G. Teubner, Stuttgart, Germany.

Brix, T., Döring, U., Reeßing, M., 2003, “Multi-Stage Modeling in the Early Phases of Design”, 14th International Conference on Engineering Design – ICED03, Stockholm, Sweden.

Brix, T., 2001, “Feature- und constraint-basierter Entwurf technischer Prinzipe”, Ph.D. thesis, TU Ilmenau, Verlag ISLE, Germany.

7. RESPONSIBILITY NOTICE

The authors are the only responsible for the printed material included in this paper.