INTERNATIONAL CONFERENCE ON ENGINEERING DESIGN
ICED 03 STOCKHOLM, AUGUST 19-21, 2003

MULTI-STAGE MODELING IN EARLY PHASES OF DESIGN

Torsten Brix, Ulf Döring, Michael Reeßing

Abstract

In early design phases three main levels of abstraction (the function, the principle and the embodiment design) can be distinguished, which describe results of the synthesis process. Suitable representations exist for these three levels of abstraction. For a continuous computer-aided product design a phase overlapping multi-stage modeling is necessary, which connects the different abstraction levels and the according representations. The aim of the paper is the presentation of a concept for multi-stage modeling and first results of an according software implementation, which supports an iterative design. The described approach is restricted to conceptual design and to the first steps in embodiment design of mechanisms and gears.

Keywords: *early phases of design, conceptual design, feature- and constraint-based design*

1. Introduction

The phase oriented view of the design process distinguishes three abstraction levels for the synthesis (Figure 1). For each abstraction level certain representations are established like functional structure, solution principles as well as embodiment design [1]. Examples are shown on the left side of Figure 1.

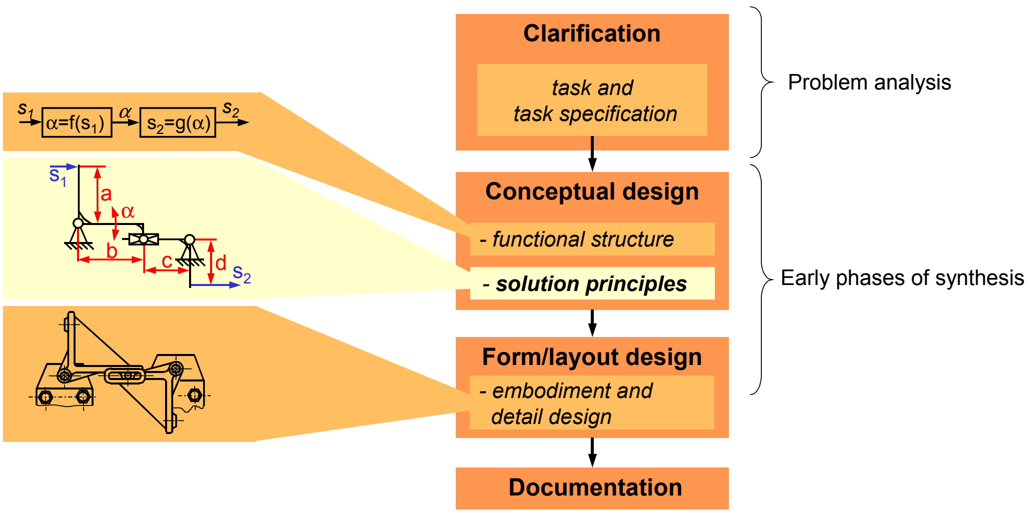


Figure 1. Simplified design process

Most computer aided design (CAD) systems start in the phase of embodiment design. Functional structures as well as solution principles are not included in such systems. In the praxis extensive experience or special tools are needed to consider the early phases of design. The usage of special tools for earlier phases (programs like Matlab, SAM, WorkingModel, Watt) can help to find a good solution principle and to determine some basic embodiment design parameters. The problem of those tools is not only that they are restricted to certain steps of the design process but also the lacking integration into the CAD working environment of the designer. Often the transfer of the results is made manually and information is lost. A further problem is that it is not possible to propagate changes made in the embodiment design back to the models in the earlier phases efficiently. Therefore iterations in the design process are not supported. This results in inconsistencies between the models in the different levels. To overcome these problems we propose a common, phase-overlapping modeling, which is supported by a catalog-oriented approach. During the design process the designer has to be able to transfer structures from a more abstract level to a less abstract level. The characteristic properties of the structure has to be kept consistent in all abstraction levels. The transfer is ambiguous (e.g. there may be different solution principles which fulfill a certain function and there is often an infinite number of possible parameter values) and may fail. Therefore in general an iterative design process is needed to obtain a good solution. During iterations the changes are propagated between the different levels via bi-directional references.

2. Feature- and constraint-based modeling

2.1 Constraint-based description

For the development of the phase-overlapping design tool the independence on the degree of complexity of structures on the three abstraction levels is very important. Therefore, a generic approach for modeling and processing is necessary. The chosen approach uses constraint-based modeling in connection with a generic constraint solver. This constraint solver is developed at the University of Ilmenau. In contrast to common numeric solvers our solver works graph-based. It supports the generation and robust handling of design variants on all levels of abstraction [2], [4].

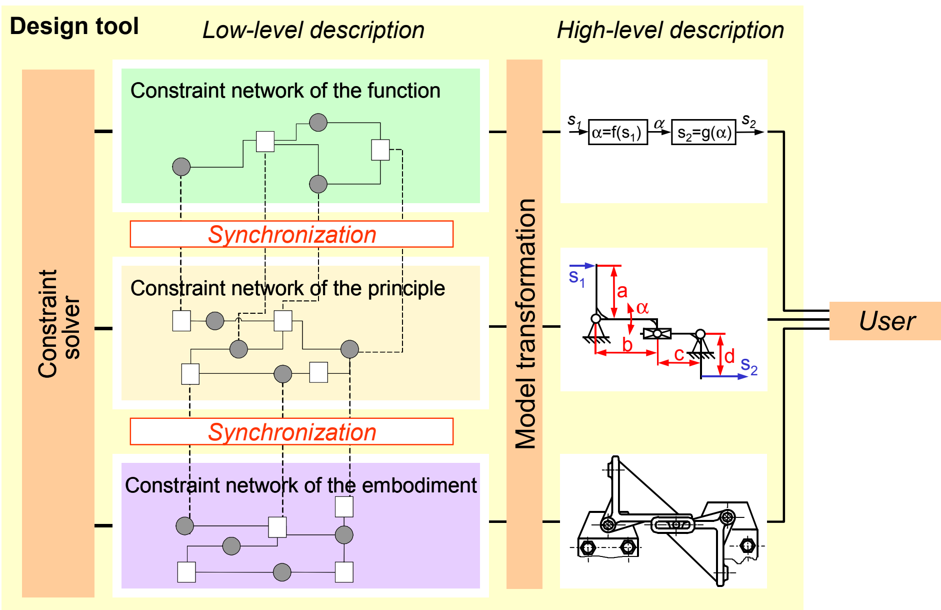


Figure 2. Simplified representation of the level-overlapping constraint network (low-level description) and the according user-oriented description (high-level description).

Constraint solving in connection with a generic constraint solver is a powerful technique for parametric design of 2D- and 3D-models and fulfils the requirements mentioned above [8]. Constraint-based models consist of parameters (scalar or vector), geometric objects (points, lines, circles,…) and constraints (with geometric semantics and some general mathematical functions), which can be defined as relations between them [6], [7]. This allows a suitable description of design objects. Functional, technological (e.g. tolerances, fits), geometric and topological properties can be integrated into one model. Figure 2 shows the application of the modelling technique to functional structures, solution principles, preliminary and detail layout of a product. Applying constraint solving to the different representations on the three abstraction levels means to simultaneously handle the following views - the intuitive, high-level description conveying the user's intent by suitable graphical representations on one side, and the constraint-based, low-level design on the other side.

2.2 Feature-based description

For the support of user-oriented modeling the constraint-based model with its interrelated parameters must be generated automatically based on the high-level description. For this purpose the feature concept can be adapted [5]. In our approach features combine data as well as methods of the high- and low-level description of each abstraction level. Thus features subsume information to describe parts of the model, which have certain semantics for designers (Figure 3).

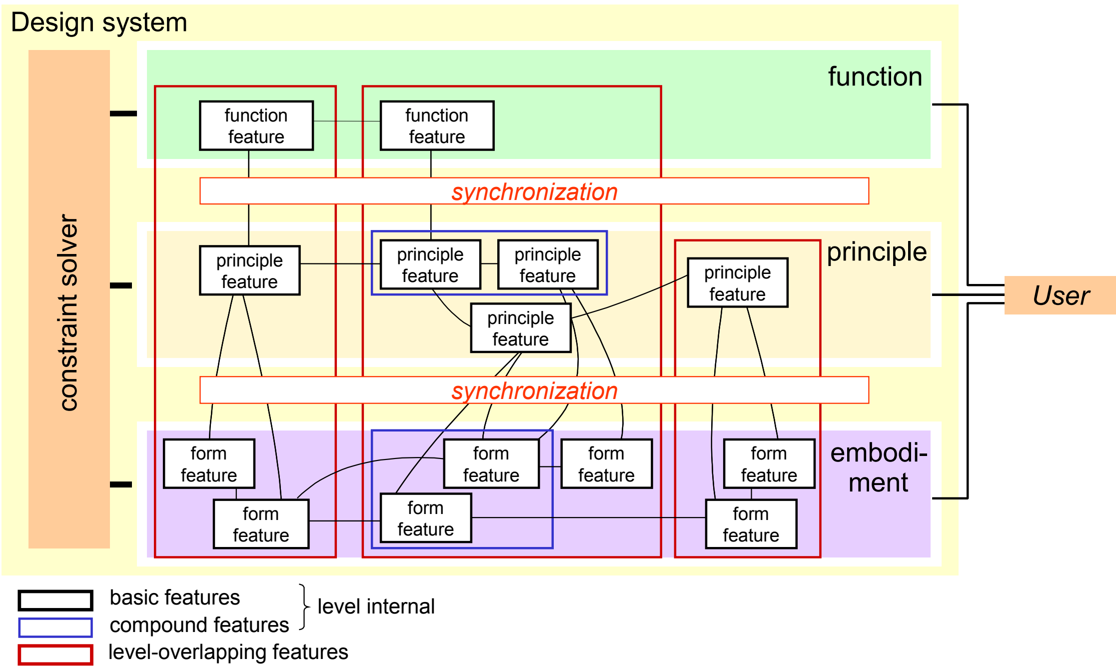


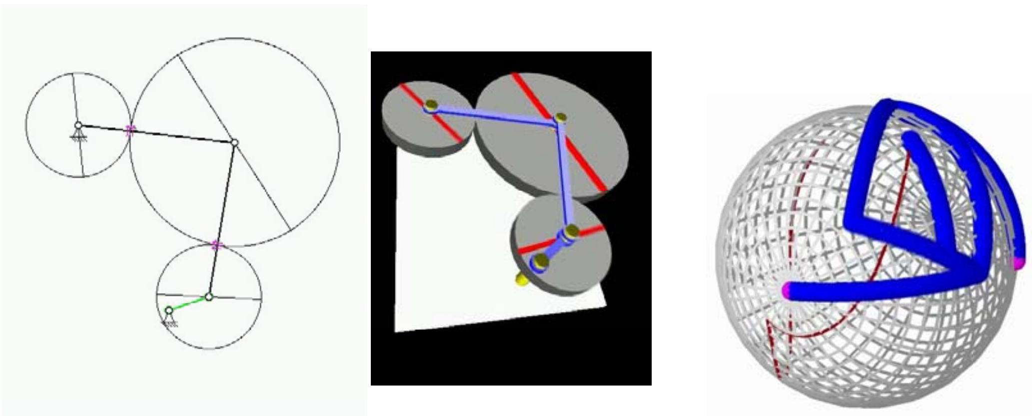
Figure 3. Relationship between basic features (combining high- and low-level description), compound features (combining basic features) and level-overlapping features (combining basic and compound features).

The constraint-based model (with its parameters, geometric objects and constraints) is generated automatically as part of each feature. Features are defined in all levels of abstraction. Within one level joining features (and according constraints) are responsible for connections between the features. Connections between corresponding features of different levels are realized by bi-directional references (dashed lines in Figure 2). Creation and deletion of features is synchronized using these references.

The design system differentiates between level-internal and level-overlapping features (Figure 3). Each level contains so-called basic features, which represent units like function elements requiring no further decomposition. Basic features can be combined in logical units. These so-called compound features can be gears or other assemblies, for instance. Besides the aggregation of basic features in compound features there are also overlapping features, which combine different levels of abstraction. The entire model representing the product is a composition of basic features, compound features and level-overlapping features.

3. Usage of a constraint solver

The constraint-based model on each level of abstraction is mapped onto a constraint graph, a so-called constraint network [9]. This allows fast degree of freedom and dependency analyses using methods from graph theory. Fast algorithms are important for interactive changes (e.g. motion simulation). For each change of certain parameters or geometric objects in the model the constraint solver generates automatically an appropriate sequence of necessary calculations [12], which ensures that the changes are propagated and all levels of the model are kept consistent. In this way the values of parameters and geometric objects, defined on the three abstraction levels, are synchronized. Figure 2 shows the constraint networks for those levels. Often mechanisms and gears can be mapped onto the plane (Figure 4a and Figure 4b). In these cases 2D constraint solving is sufficient, even though the visualization in embodiment design is 3D. To handle spherical mechanisms (Figure 4c) as well as special properties in the embodiment design 3D constraint solving is involved.



a) b) c)

Figure 4. Mechanisms and gears modeled in MASP (see section 5):
a) Solution principle of a geared linkage,
b) Embodiment design of a geared linkage shown in Figure 4a,
c) Solution principle of a spherical crank-rocker mechanism.

4. Model synchronization between levels of abstraction

In addition to the description of the model structure on one level a mapping between the parameters of corresponding features on the three abstraction levels is necessary. Invariant and easy transformable information is particularly suitable for this purpose. For instance, the transition of a technical principle into its embodiment can be done using characteristic axes, distances and angles, which can be found in both model representations (Figure 5).

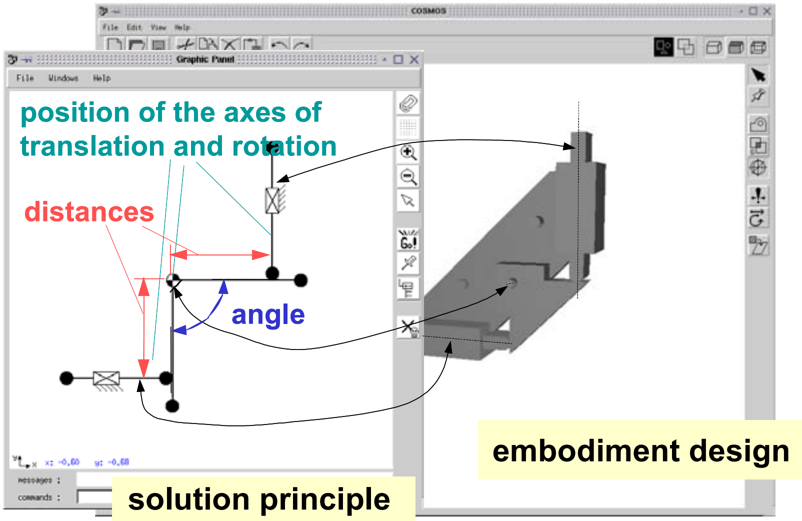
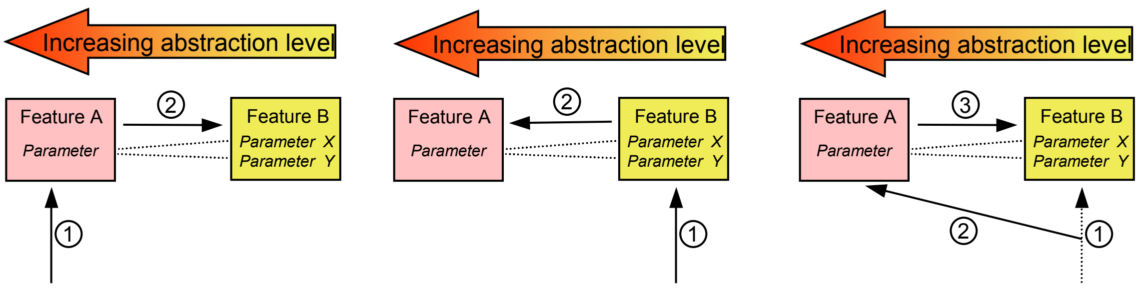


Figure 5. Solution principles as basis for the embodiment design.

These parameter mappings as well as the references describing the structure have to operate bi-directionally in order to propagate changes from one level to the other levels. This approach ensures the consistency of the entire model. The amount of information increases from the functional structure to the embodiment design. A feature on a more concrete level contains all the information of the corresponding features on more abstract levels. Therefore some of its parameters are also relevant for these corresponding features. The technical principle of a bar, for example, has a parameter length. The according embodiment design with a circular cross section has also a length and additionally a diameter. While the diameter may be changed without affecting the technical principle, the length has to be synchronized between both models. Figure 6 shows possible scenarios for parameter changes on a certain abstraction level and how the other levels are affected.



(a) Direct change (1) of a parameter in feature A causes the change (2) of associated parameters in Feature B.

(b) Direct change (1) of parameter Y in Feature B with propagation (2) to an associated parameter in Feature A (Problem: possible ambiguities).

(c) Indirect change (1) of a parameter Y in Feature B via the change (2) of the according parameter in Feature A which in turn affects the parameters in Feature B (3).

Figure 6. Possible scenarios for parameter changes.

5. Catalog-oriented design

To achieve an easy usability an assistant software for catalog-oriented design is used. It supports the user in modeling of the intended product attributes (e.g. layout/form, material, technological properties) using predefined solution elements (Figure 7 and Figure 8) [10], [11].

Such solutions are features, which represent components, assemblies and systems. They should be available in different levels of description and different quantitative variants to perform functions within a certain range of parameters. The combination of solution elements produces the desired product variants. This procedure based on a logical sequence of design steps is called configuration [3]. Products with a given functional structure can be configured directly by parameter specification, choice of components and layout generation.

Based on models, which are defined in the functional or principle stage, suitable solutions for subsequent model levels can be generated. Necessary bi-directional references between the features of the three model levels are added automatically during the generation. This allows iterations during the design process. During operations like creation, modification and deletion of one or more solution elements at a certain abstraction level the relations to other solution elements are automatically considered (changes in lower levels of abstraction are propagated to higher levels and vice versa).

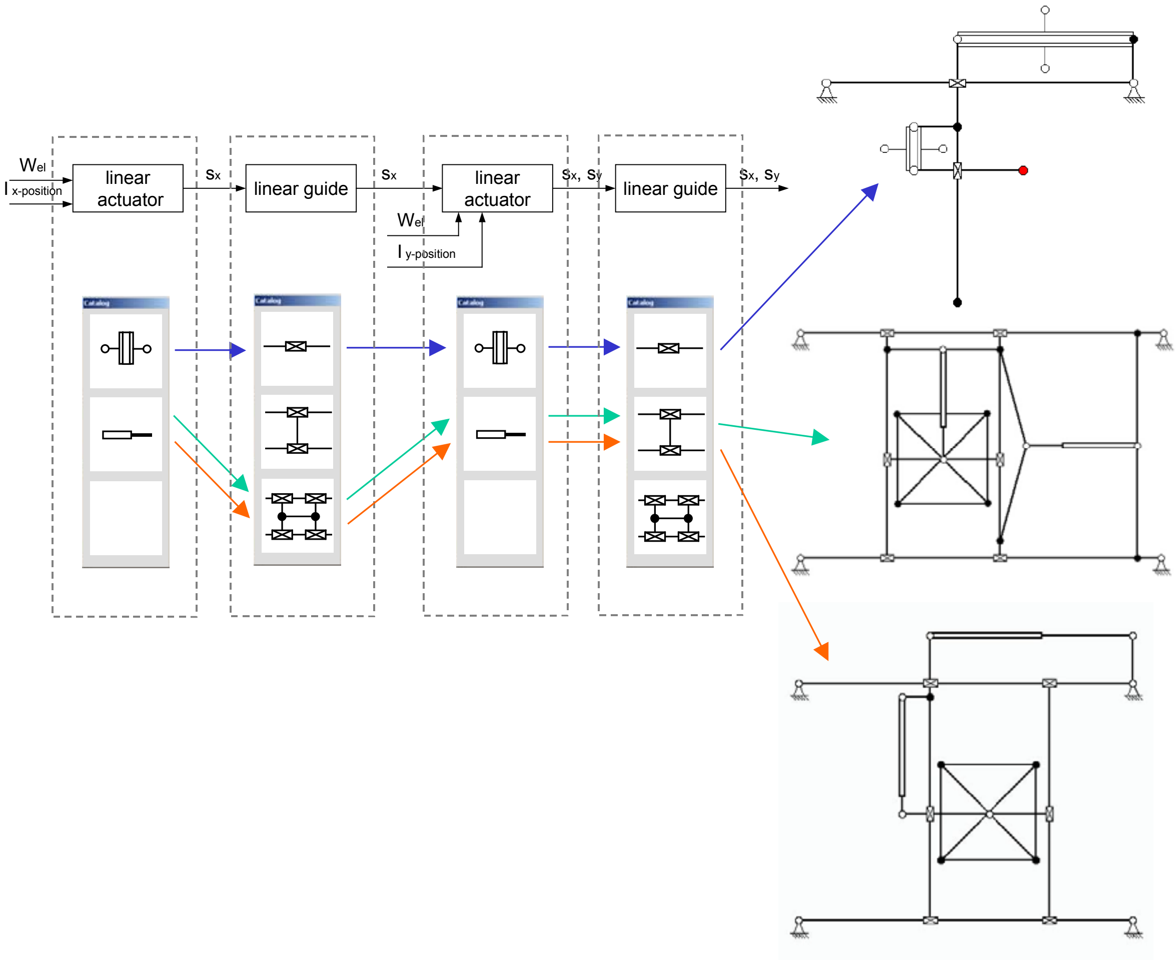


Figure 7. Determination of solution variants by configuration which is based on a functional structure.

In this way the design system enables the user to create alternative solutions, which may be analyzed, simulated and optimized regarding several domains of behavior. The design system provides additional improvement of the modeling process by consideration of information to complete the model. For example, it is necessary to consider material properties in the dimensioning process of the embodiment which is based on solution principles. Material properties in combination with stress analyses facilitate generation of different solution variants depending on which information is given. As an example, from the specification of the material and applied stress the dimensions of a part can be calculated. In reverse a suitable material can be determined by the specification of the part dimensions and the material stress.

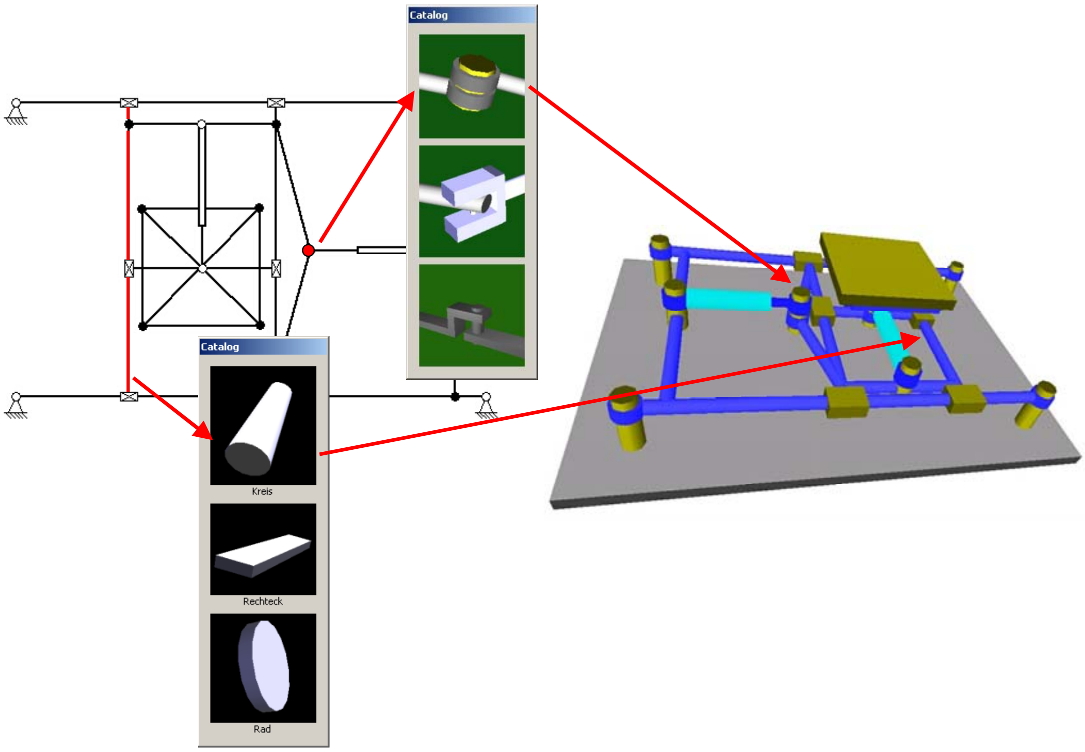


Figure 8. Determination of solution variants by configuration which is based on a solution principle.

The design system includes solution catalogs for product configuration. Among other information they contain relations between the solutions on different abstraction levels. This defines a level spanning hierarchy for the product development (Figure 9).

It is up to the user to decide on which level of abstraction the design process starts. For models existing as a functional structure or as a technical principle the design system facilitates the creation of according solutions for the respective succeeding level. To support this process an assistant system is included. Using information from the catalog hierarchy as well as context-sensitive and task specific restrictions the assistance system offers a number of fitting solutions for selected basic and compound features. For example, the function element “amplification” is implemented by a range of principle elements like levers and gears. In addition the set of possible solutions is reduced by requirements of task, compatibility to adjacent elements, results of dimensioning calculations, material related issues, etc.

Some of the ideas described in the previous sections have already been implemented in an application called MASP (program for Modeling and Analyses of Solution Principles, Figure 10). The interactive modeling of solution principles is done by selecting symbols in the context of chosen instruction (e.g. create, delete, modify). For the first steps in embodiment design predefined form elements exist in the mentioned design system (Figure 8).

In this way it is easy to configure models of planar or spherical mechanisms and gears interactively with the aid of predefined solution elements. Solution variants can be also determined interactively. The variants of the model can be analyzed, simulated und optimized related to different properties (e.g. motion simulation using the constraint solver or calculation of static/kinematic quantities by additional algorithms, Figure 11b).

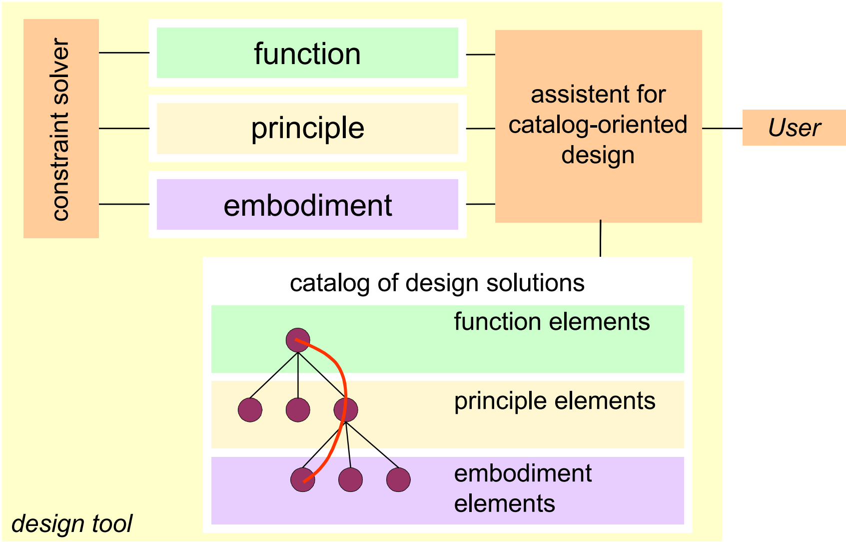


Figure 9. Catalog-oriented design.

MASP enables the user to immediately test the functionality of the current design concept, for instance, by interactive mouse drags or by applying further calculations (e.g. kinematics or static calculations) based on the evaluated constraint model [7].

The interplay of the different description levels is illustrated from Figure 7 to Figure 11. For example the user modifies the model interactively by dragging a joint. This information will be saved in the data part of the joint (see section 2). Based on the current parameters and positions in the low-level description the constraint solver computes the new positions of all connected geometric entities as well as non-geometric data. Furthermore other calculation modules will be used to recalculate dependent data, for instance to determine velocities and accelerations (Figure 11b). After this, the updated high-level description is used to modify the representation on the screen, for instance the symbol of the spring, which may include a visualization of the force. That may be a value in the functional structure and vectors in the visualization of the solution principle as well as the embodiment design.

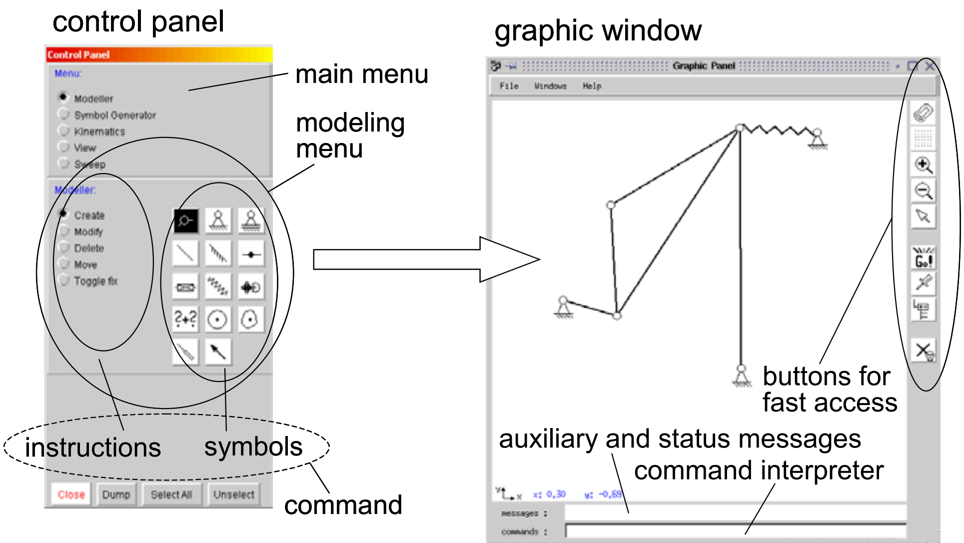


Figure 10. Design system MASP

a)

b)

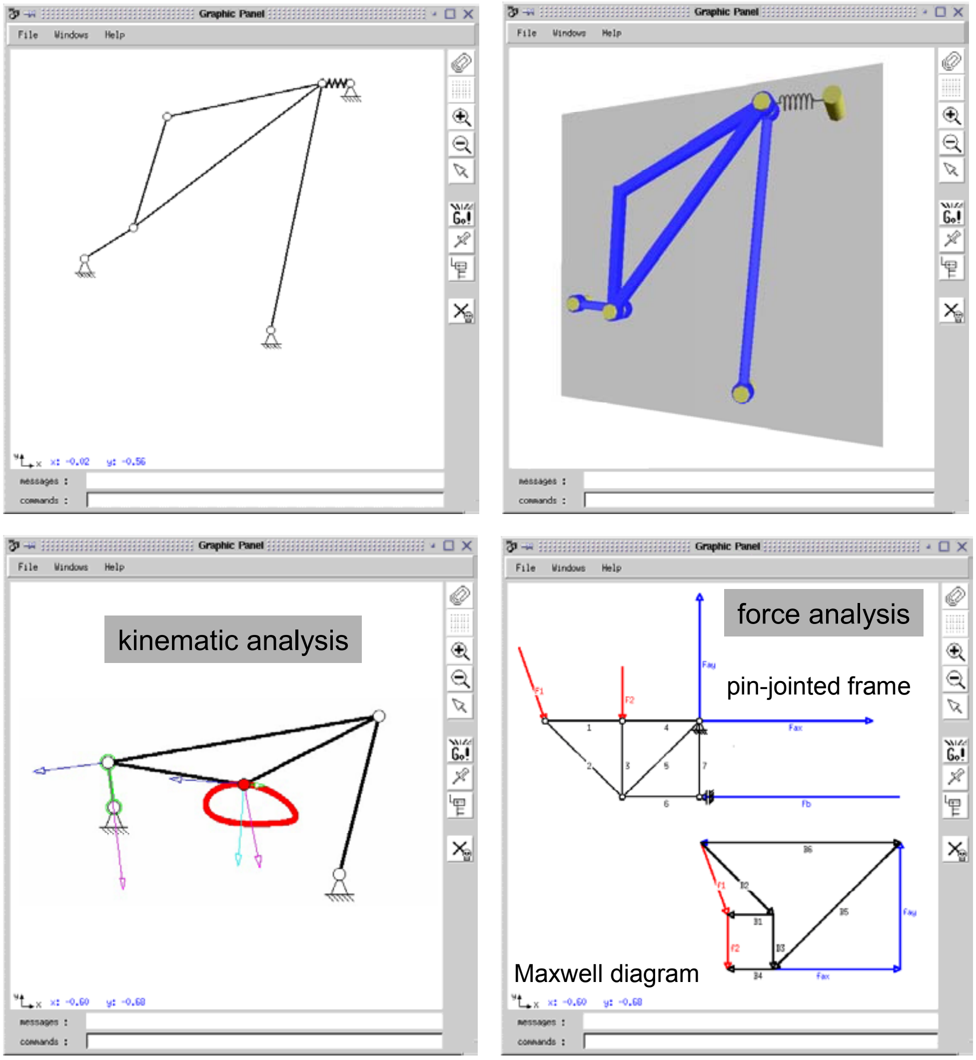


Figure 11. Design system MASP for solution principles:
a) Interactive simulation of motion by mouse dragging in 2D and 3D,
b) Calculations based on the evaluated constraint model.

6. Conclusion and Further Work

The paper presents a suitable concept and new ideas for a computer-based design system, which supports a phase-overlapping multi-stage product design in early design phases. The different models for each stage are integrated into one design system. A feature- and constraint-based model description is proposed to synchronize the different levels of abstraction and to keep the model consistent. The approach is partially implemented in a system, which supports the design of planar and spherical mechanisms [2], [4], [6], [7]. It reflects our experiences in the field of feature- and constraint-based modeling of solution principles, embodiment design as well as the transition between both.

Further work is focused on the implementation of the functional structure and the according catalog.

The work is supported by Deutsche Forschungsgemeinschaft (German Research Foundation).

References

[1] Pahl, G. and Beitz, W., "Engineering Design - a systematic approach", Springer-Verlag, 1988.

[2] Brix T., Brüderlin B., Döring U. and Höhne G., "Feature- and constraint-based design of solution principles", Proceedings of ICED '01, Design Research – Theories, Methologies and Product Modelling, Glasgow, 2001, pp. 613-620.

[3] Riitahuhta A. and Pulkkinen A. eds., "Design for Configuration, A Debate based on the 5th WDK Workshop on Product Structuring", Springer-Verlag, Berlin, 2001.

[4] Brüderlin B., Döring U., Klein R. and Michalik P., "Declarative Geometric Modeling with Constraints". Proceedings of CAD 2000, Berlin, 2000, pp. 377-393.

[5] Shah J.J. and Mäntylä M., "Parametric and Feature-based CAD/CAM", John Wiley & Sons, New York, 1995.

[6] Brix T., Brüderlin B. and Döring U., "Conditional Constraints in Conceptual Design", Proceedings of 32st ISATA, Wien, 1999, pp. 441-448.

[7] Brix T., Brüderlin B., Döring U. and Höhne G.: "Representation and Analysis of Solution Principles". Proceedings of CAD 2002, Dresden, 2002, pp. 45-55.

[8] Roller D. and Brüderlin B. eds.., "Geometric Constraint Solving and Application", Springer-Verlag, Berlin, 1998.

[9] Hsu C. and Brüderlin B.: "A hybrid Constraint Solver Using Exact and Iterative Geometric Constructions". CAD System Development: Tools and Methods, Springer-Verlag, Berlin, 1997, pp. 265-279.

[10] Koller R., Kastrup N., "Prinziplösungen zur Konstruktion technischer Produkte", Springer-Verlag, Berlin, 1998.

[11] Roth K., "Konstruieren mit Konstruktionskatalogen", Springer-Verlag, Berlin, 2000.

[12] Brix T., Döring U. "Constraint-basierte Berechnung kinematischer Geometrien" in Proceedings of Dresden Symposium Geometry: constructive & kinematic, to appear in November 2003.

Torsten Brix, Michael Reeßing
Faculty of Mechanical Engineering
Engineering Design Group

torsten.brix@tu-ilmenau.de
michael.reessing@tu-ilmenau.de

Ulf Döring
Faculty of Computer Science and Automation
Computer Graphics Group

ulf.doering@tu-ilmenau.de

Technical University of Ilmenau, PO Box 100565, 98684 Ilmenau, Germany
Phone: +49 3677-69-1211, Fax: +49 3677- 69-1285