Function-oriented Configuration of Products by means of Feature and Constraint-based Modeling

Prof G Höhne and Dr T Brix

Keywords: feature-based design, constraint-based modeling, modularization, configuration

1 Introduction

In the early design phases three main levels of abstraction can be distinguished, which describe the function, the principle and the embodiment (layout/form). Suitable representations (e.g. function structure, solution principles, preliminary layout or detail layout) exist for these three levels of abstraction [1]. For a continuous computer-aided product design a phase overlapping multi-stage modeling is necessary, which connects the different levels and the according representations. The aim of the paper is the presentation of the concept and an exemplary implementation of a convenient software application, which supports an iterative design (changes in lower levels of abstraction are propagated to higher levels and vice versa). The described approach based on a function-oriented configuration. For this four procedures of configuration of products will be discussed. The exemplary implementation is oriented to the conceptual design (determination of function structure and solution principle) and the first steps in embodiment design (development of preliminary layout and form) of technical products with mechanical components.

2 Feature- and constraint-based modeling

For the development of a generally applicable, phase overlapping design tool the independence of specific structures and of the degree of complexity on the different levels is very important. Therefore, a generic approach for modeling and processing must be used. Constraint solving in connection with a generic constraint solver is a powerful technique for parametric design of 2D- and 3D-models and fulfills the designated requirement [2]. The models are described by parameters, geometric elements and constraints between them. This modeling technique is suitable for modeling functional structures, solution principles, preliminary and detail layout of a product (Figure 1). In this way functional, technological, geometric und topological properties can be integrated into one model.

It is possible to integrate non-geometric quantities into the constraint-based model by means of equation constraints. An example on the three main levels of abstraction is shown in Figure 1, where a piezo element causes a translation according to a given voltage. In addition Figure 1b shows the constraint based model and the constraint graph of the solution principle as example for a suitable constraint-based model description. Table 1 contains some examples for geometric and non-geometric constraints, which will be used to generate constraint-based models like shown in Figure 1b (refer to [5] for a more detailed description of the modeling syntax).

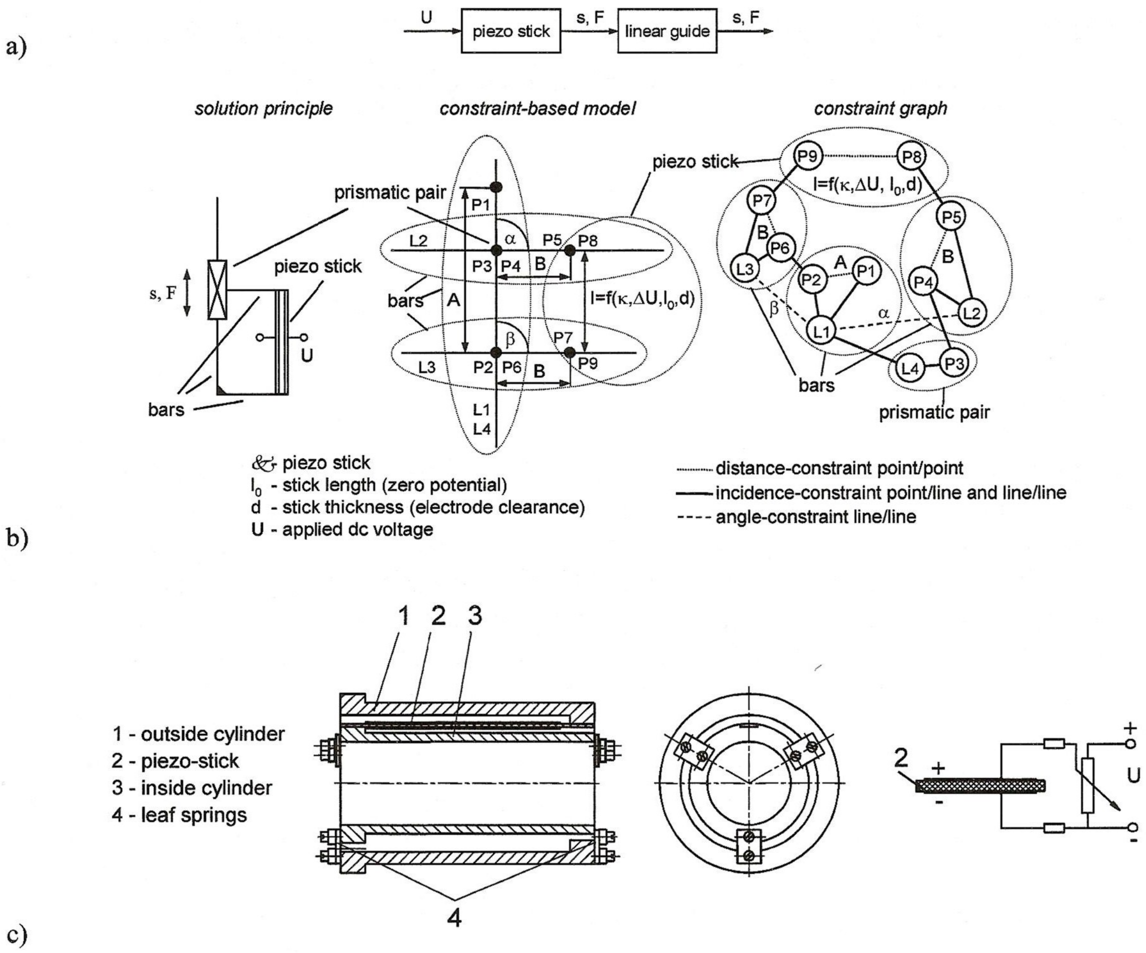
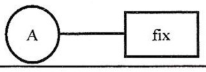
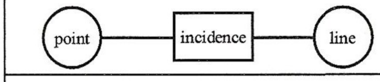
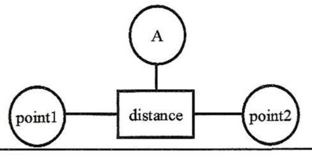
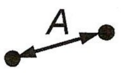
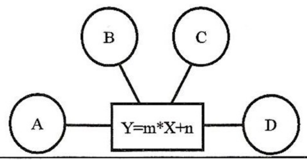


Figure 1. Model of a translation element
a) functional structure
b) solution principle with the corresponding constraint model and graph
c) detail layout

Table 1. Examples for geometric and non-geometric constraints

constraint graph (□- constraint, ○-object)	comments
	A=const. (fix constraint)
	point lies on the line (incidence constraint)
	 two points with a certain distance A (distance constraint)
	A=B·C+D (equation constraint)

Often mechanisms and gears for example can be mapped onto the plane. In these cases, 2D constraint solving is sufficient, even though the visualization in embodiment design is 3D. To handle spherical mechanisms 3D constraint solving is used. The constraint-solver is developed at our University [6]. It supports the generation and robust handling of design variants on all levels of abstraction.

Applying constraint solving to the development of models means to simultaneously handle the following views - the intuitive, high-level description conveying the user's intent by symbols, pre- and user-defined elements or parts , on one side, and the constraint-based, low-level design on the other side (see Figure 2).

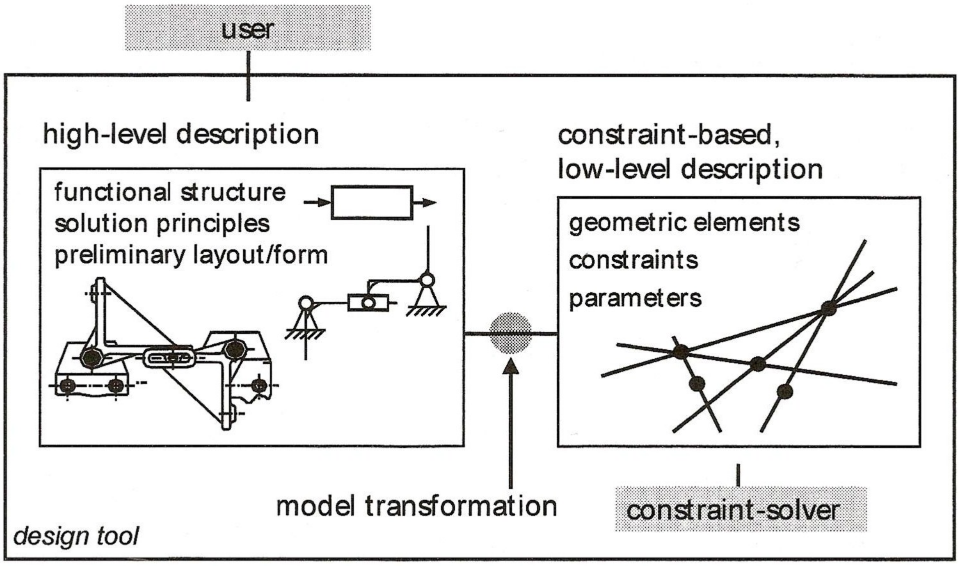


Figure 2. Connection between high-level and constraint-based description

For a suitable combination of high-level and low-level description we employ the feature concept [3]. Features are a subsumption of both descriptions. Features combine data and methods of the two levels of description as one entity (see Figure 3).

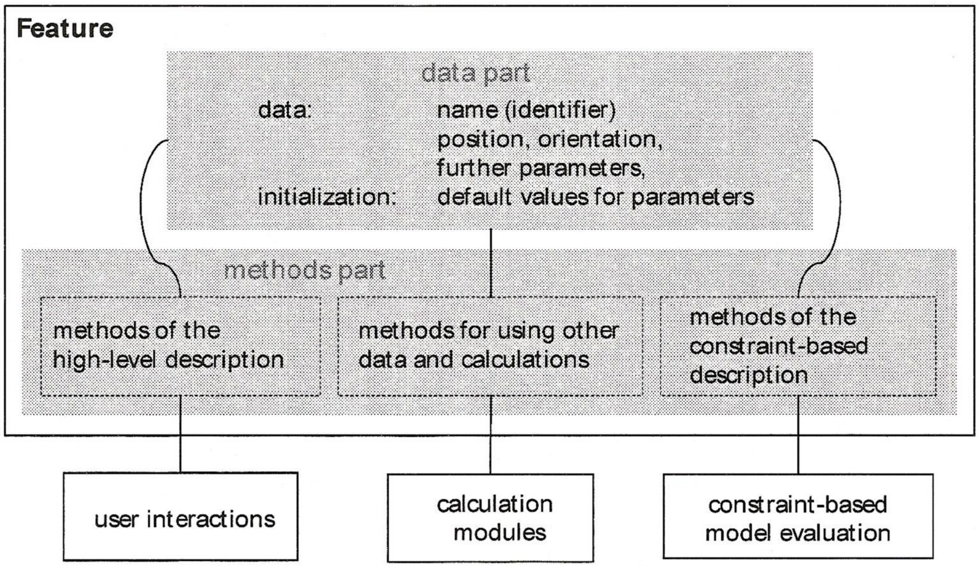

Feature
data part
data:
name (identifier)
position, orientation,
further parameters,
initialization:
default values for parameters
methods part
methods of the high-level description
methods for using other data and calculations
methods of the constraint-based description
user interactions
calculation modules
constraint-based model evaluation

Figure 3. General structure of a feature

In the data part, common information (e.g. name, position and orientation) and symbol specific information (e.g. shape and IDs for the constraint solver) are stored. Data will be manipulated using the methods defined in the methods part. Three groups of methods can be distinguished. The methods of the high-level description implement user interaction, drawing of symbols, pre-, user-defined elements and parts. The low-level methods generate a suitable constraint-based description. Furthermore, they implement the interface to the constraint solver for geometric evaluation and data transfer. The third group provides an interface to other necessary calculation modules, e.g. for kinematics or static evaluations. The shared data concept and according update mechanisms allow the synchronization between the different descriptions in the feature.

3 Design phases overlapping models

Figure 4 shows the design process and the relationship between functional, principle and part structure. Here, solution principles consist of principle elements and couplings (e.g. joints). This distinction is important when solution principles are used as basis for part design (couplings become connections)[1].

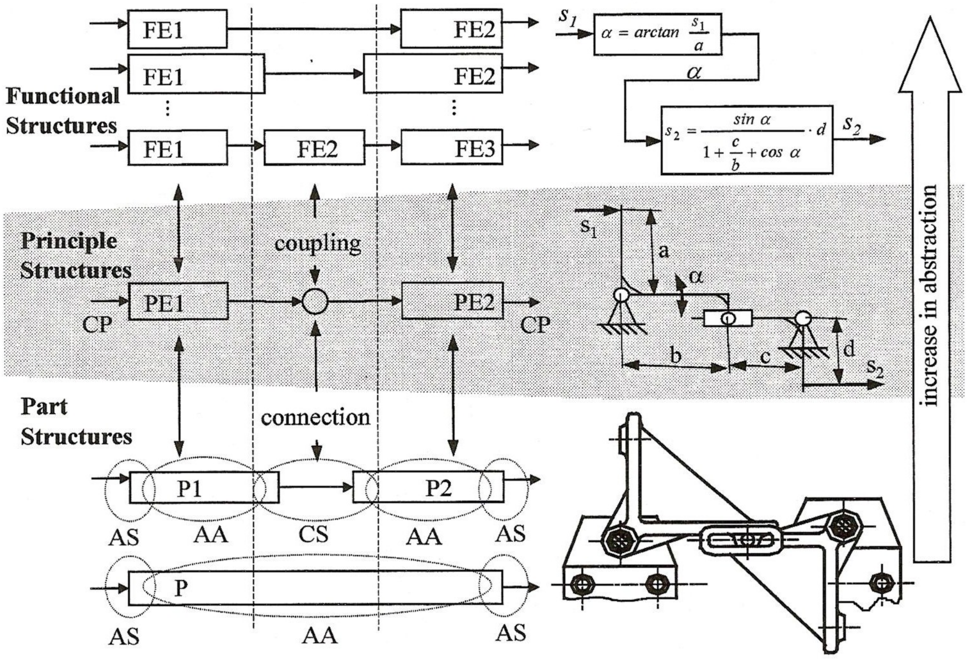


Figure 4. Relationship between functional, principle and part structure (FE = function element, PE = principle element, P = part, CP = couple point, AS = active surface, CS = contact surfaces, AA = active area)

Design tool internally the constraint-based model on each level of abstraction is mapped onto a constraint graph respectively constraint network (Figure 5). That allows fast degree of freedom and dependency analyses using methods from graph theory. For each change of certain parameters or geometric objects in the model the constraint solver generates automatically an appropriate sequence of necessary calculations, which ensures, that the changes are propagated and all levels of the model are kept consistent. That way the values of parameters and geometric objects, defined in the different levels, are synchronized by use of references. Figure 5 shows the constraint networks for different levels. Dashed lines between the networks clarify the references for the synchronization.

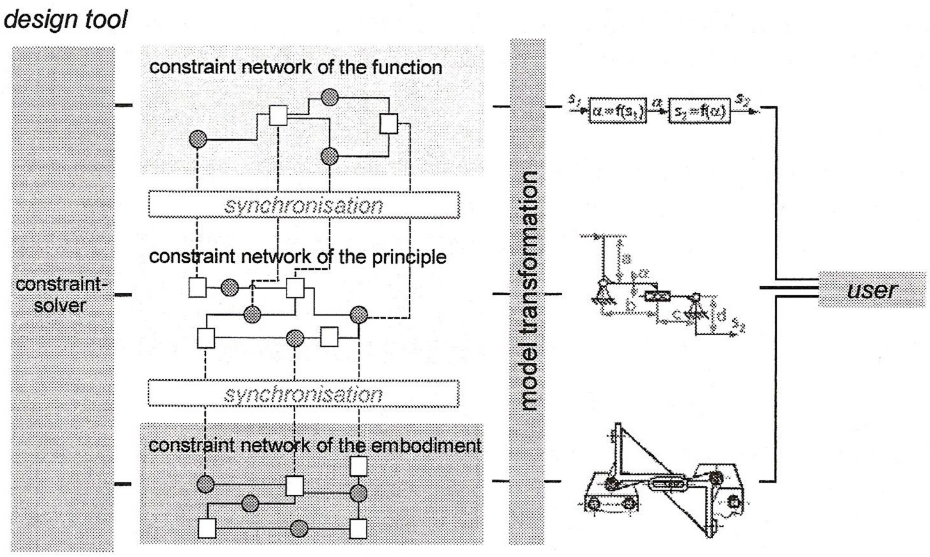

design tool
constraint network of the function
synchronisation
constraint network of the principle
constraint-solver
synchronisation
constraint network of the embodiment
model transformation
user

Figure 5. Phase overlapping constraint network

Predefined solution elements (features, which represent components, assemblies and systems), which integrate sections of the complete constraint network (Figure 3), are supplied to support user-oriented modeling. 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 (Figure 6) and are termed function, principle and form feature. On each level of abstraction exist basic features, which represent units like function elements. Basic features can be combined to logical units. These compound features represent gears or mechanisms for example. According to this, phase-overlapping features contain defined model properties on different levels of abstraction. The complete model consists basis, compound and phase-overlapping features.

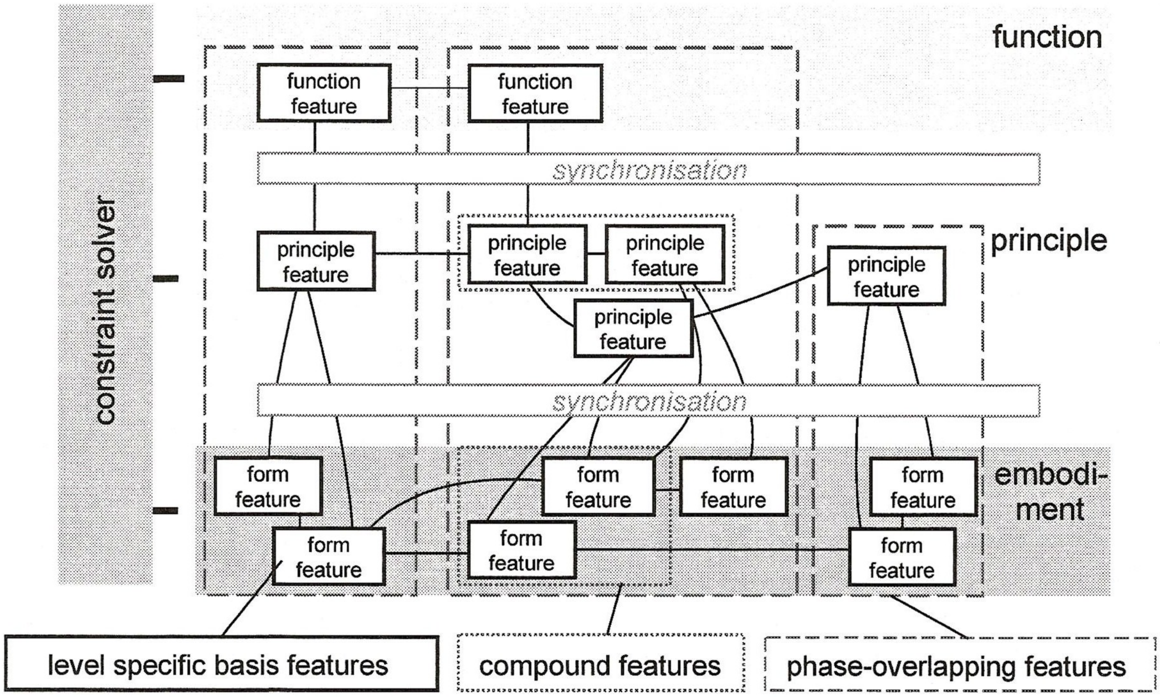


Figure 6. Basis, compound and phase-overlapping features in relationship to the constraint solver

Within one level joining features (and according constraints) are responsible for connections between the elements. Connections between corresponding features of different levels are realized by bi-directional references (Figure 5). Creation and deletion of features is synchronized using these references. Features supply level specific attributes (data) and different methods. These methods are responsible for the interactive generation and modification of the model and its graphical representation. Furthermore, they allow the coupling to calculations like kinematical analyses, motion simulations or force analyses (Figure 3).

4 Approaches of configuration procedures for modular systems of products

Equipment for measurement, fabrication, assembling processes, cars, robots and many other machines and devices have modular structures to facilitate production, installation, application, service and recycling of these products. This kind of structure has also the advantage that the product can be configured conform to the costumer requirements.

The design process of such products starts with the function structure. Each subfunction can be carried by various prepared modules. A module should be available in different levels of description and different quantitative variants to perform functions within a certain range of parameters. The combination of the modules produces the desired product variants. This procedure, based on a logical sequence of design steps, is called configuration [4][7].

The configuration procedure depends on the type of function structure. Products with a given function structure can be configured directly with the steps: parameter specification, choice of components and layout generation (Figure 7).

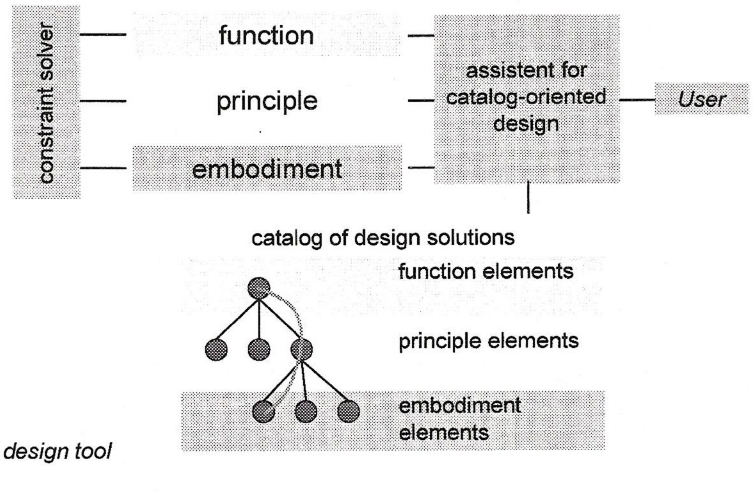


Figure 7. Catalog- and function-oriented configuration of modular products

In relation with the mode to establish the functional structure can be distinguished four configuration procedures:

1. **Configuration by using prepared function structures**, stored in a catalog (selection procedure). Figure 1 shoes a simple function structure of a linear positioning system. It forms as base for configuration of other variants with different types of linear actuators (electro magnet, linear motor, magneto-strictive drive) and guides (as sliding; rolling, aerostatic or spring guides).

2. **Configuration by combination of sub-structures.** From the simple one coordinate function structure in Figure 1 can be configured a two (or also multiple) coordinate positioning system by a serial arrangement (Figure 8).

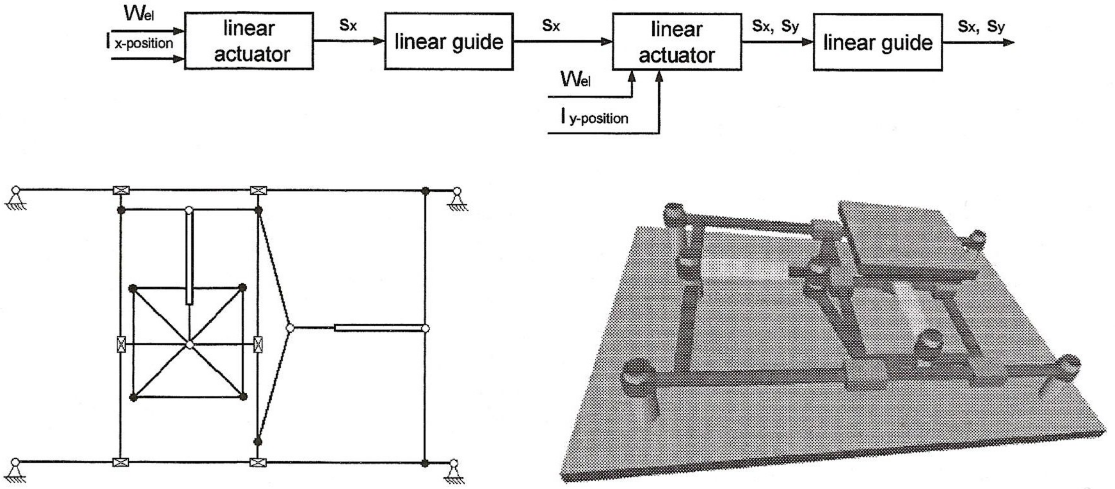


Figure 8. Configuration of a two coordinate positioning system by combination of two linear units from Figure 1

3. **Configuration by variation of given function structures.** Each functional structure can be modified by changing the arrangement of the function elements. From the initial functional structure in Figure 8 are derived variants in Figure 9a and Figure 9b.

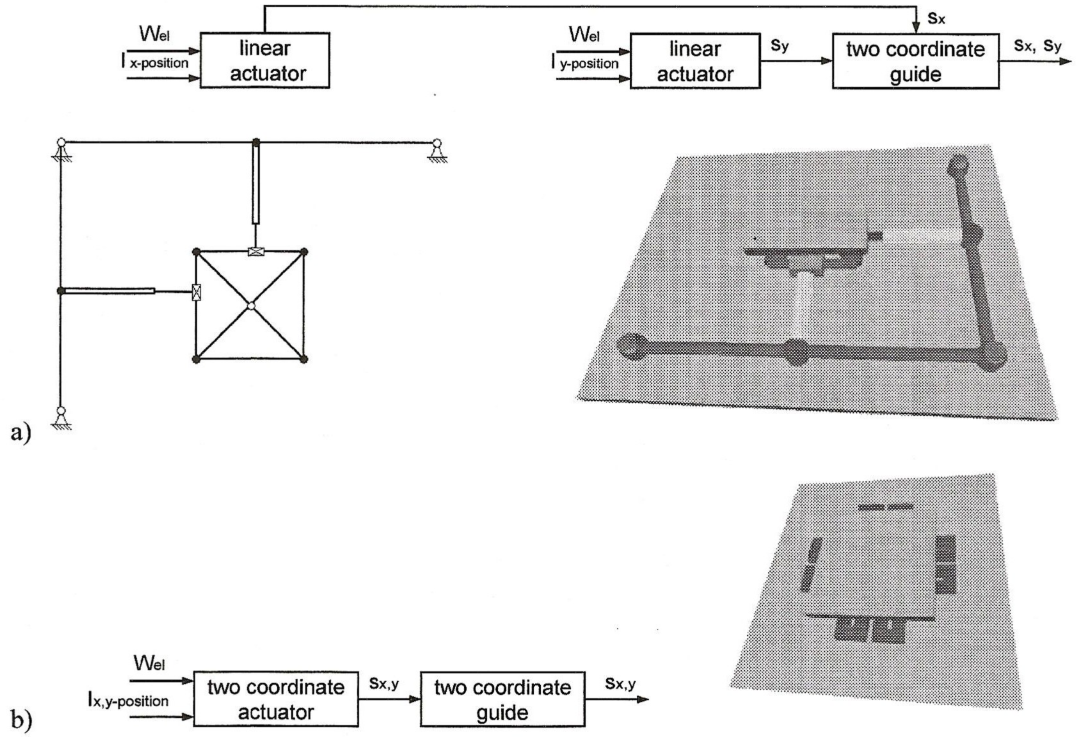


Figure 9. Configuration by variation of given function structure and possible configurations as solution principle and embodiment design (using design system MASP, see section 5)
a) variation using a two-coordinate guide; b) variation using two-coordinate actuator and guide

4. **Configuration by breaking down a complex function structure.** The base of this procedure is a maximum function structure representing a product family with all possible sub functions (platform concept). In relation with the actual task or application the structure is modified by elimination of not necessary function elements (Figure 10).

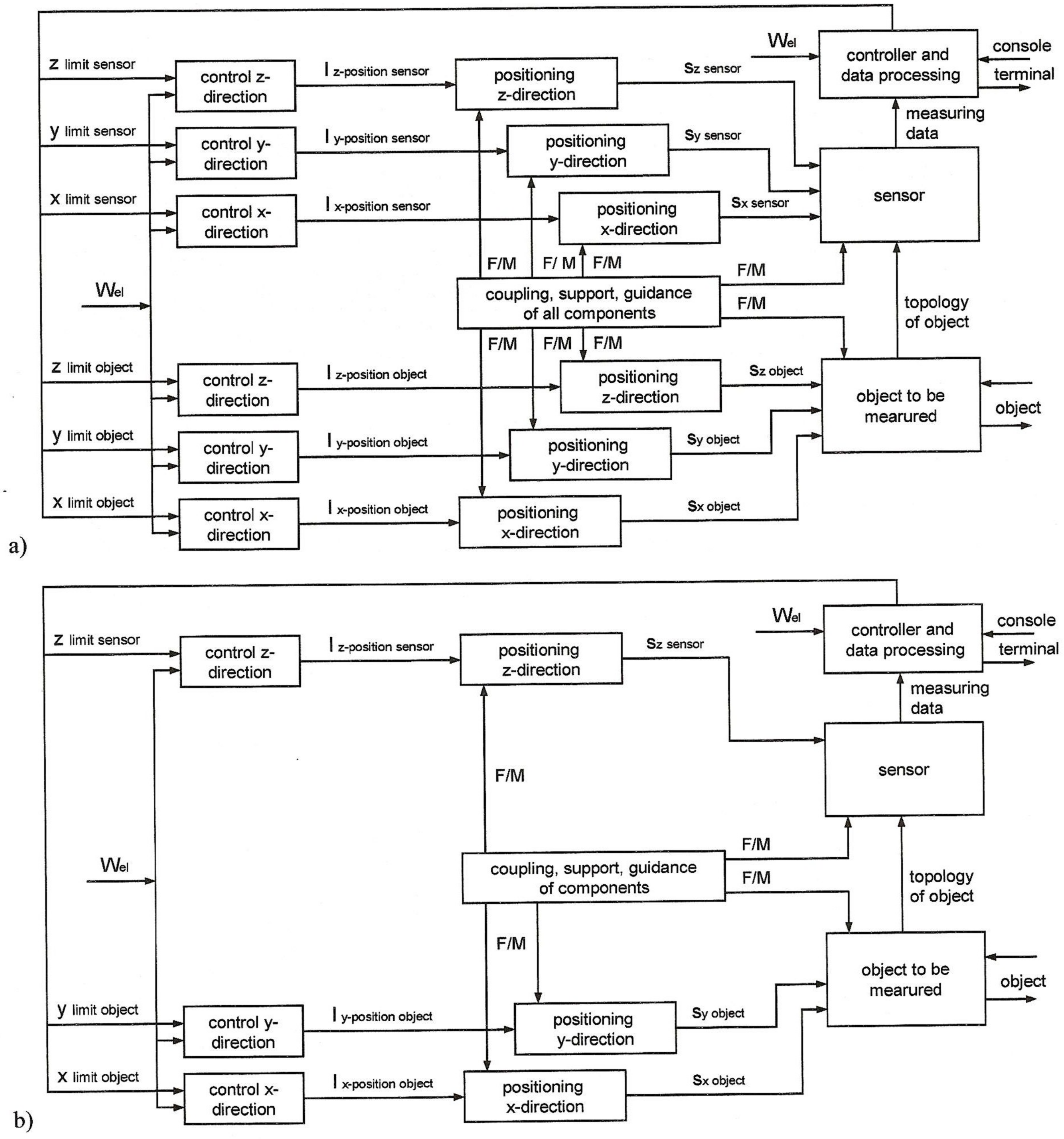


Figure 10. Configuration by breaking down a complex function structure
a) complete function structure of a coordinate measuring system (translation only)
b) derived and reduced function structure
(F-force, M-moments, I-information, s-stroke, W-electrical power)

5 Implementation

The ideas described in the previous sections have been implemented in an application called MASP (program for Modeling and Analyses of Solution Principles). 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 exist predefined form elements in the mentioned design system. An example of a modeling sequence for a crank-rocker mechanism is given in Figure 11a. Examples of embodiment design in MASP are shown in the Figure 8 and Figure 9.

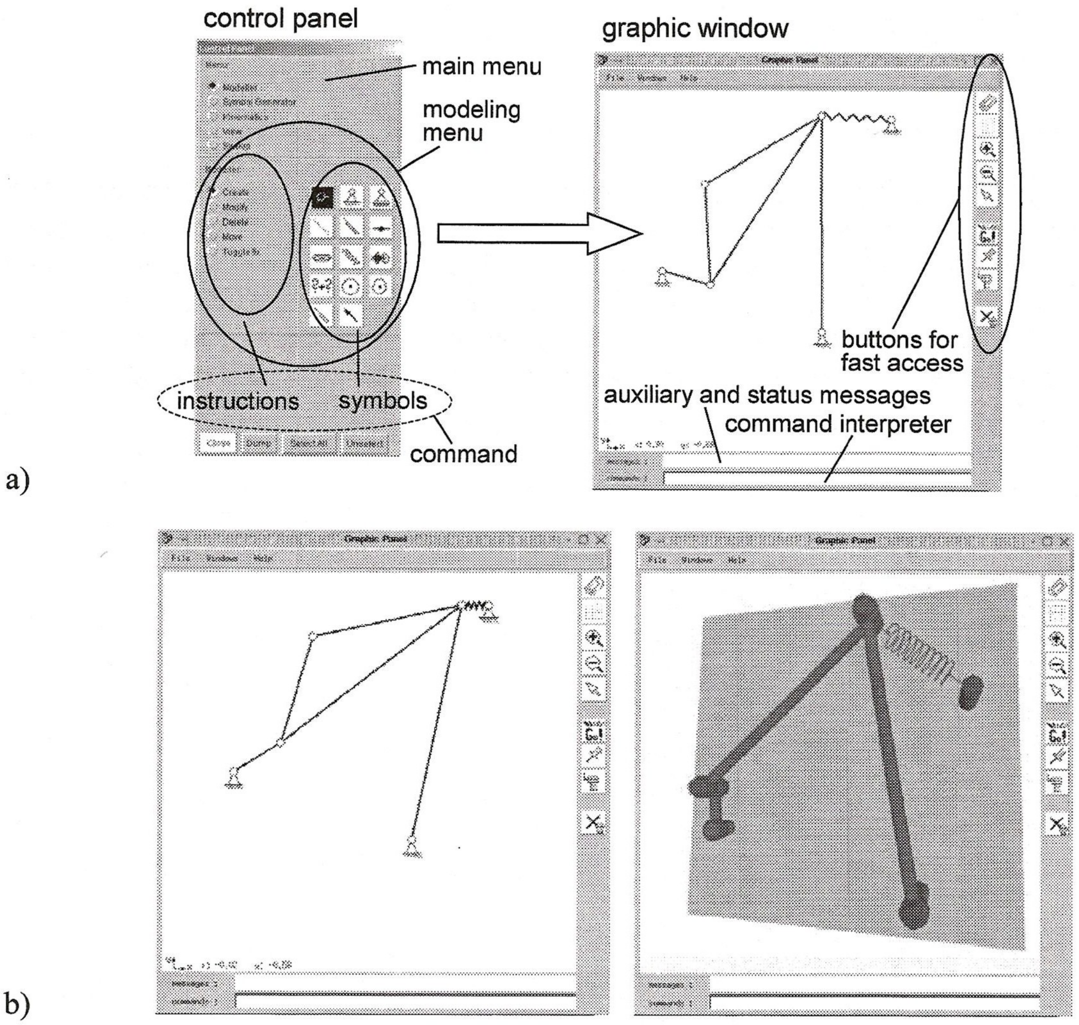


Figure 11. Design system MASP for solution principles
a) interactive modeling of a crank-rocker mechanism
b) interactive simulation of motion by mouse dragging in 2D and 3D

During interactive high-level modeling, a low-level description by constraints is generated automatically. MASP enables the user to test immediately 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 [5].

The interplay of the different description levels is illustrated in Figure 11. The user modifies the model interactively by dragging a joint. This information will be saved in the data part of the joint (feature, see Figure 3). 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 physical forces. 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.

It is easy to configure models of planar and spherical mechanisms and gears interactively with the aid of predefined solution elements respectively features. Solution variants can be determined interactively too. The variants of the model can be analyzed, simulated und optimized concerning different properties.

6 Conclusion and Further Work

We developed the approach and a first implementation called MASP, an application which supports the conceptual design phase and the first steps in embodiment design. The feature concept was used to combine different description levels on different levels of abstraction. It supports a function-oriented configuration of modular products. The constraint-based model allows to perform various analyses to find a solution that fulfils the requirements. A bi-directional model transformation is used for the transition between 2D-solution principles and 3D solid models. In future work, additional solution elements and calculation methods as well as visualization of calculation results will be integrated into MASP. The presented work follows from a research project, which is sponsored by German Research Foundation (DFG).

References

[1] Pahl, G.; Beitz, W.; Wallace, K., "Engineering Design", Springer-Verlag, Berlin, 1999.

[2] Roller, D.; Brunet, P., "CAD-System Development", Springer-Verlag, Berlin, 1997.

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

[4] Höhne, G.: "Configuration of Modular Products by Means of Mechatronic Components", Proceedings of the European Conference in Focused Aspects of Mechatronics, Ilmenau 1997, pp. 121-130.

[5] Brix, T; Brüderlin, B.; Döring, U.; Höhne, G.: „Feature- and constraint-based design of solution principles“, Proceedings of the 13th International Conference on Engineering Design – ICED01, Glasgow, 2001, pp. 613-620.

[6] C. Hsu and B. Brüderlin, „A hybrid geometric constraint solver using exact and iterative geometric constructions“, CAD Tools and Methods for Design System Development, D. Roller and P. Brunet Eds., Springer-Verlag, Berlin, 1997.

[7] Riitahuhta, A.; Pulkkinen, A. (Eds.): Design for Configuration, A Debate based on the 5th WDK Workshop on Product Structuring, Springer-Verlag, Berlin, 2001.

Prof G Höhne and Dr T Brix
Technical University of Ilmenau
Dept. of Mechanical Engineering
Engineering Design Group
PO Box 100565, 98684 Ilmenau, Germany
Phone: +49 3677-69-2473,
Fax: +49 3677 69 1259
E-mail: guenter.hoehne@mb.tu-ilmenau.de and torsten.brix@mb.tu-ilmenau.de