INTERNATIONAL CONFERENCE ON ENGINEERING DESIGN, ICED'07
28 - 31 AUGUST 2007, CITE DES SCIENCES ET DE L'INDUSTRIE, PARIS, FRANCE

A HETEROGENEOUS MODELLING APPROACH FOR DOMAIN ALLOCATION IN MECHATRONICS

Sebastian Jansen[1] and Ewald G. Welp[2]
[1]Robert Bosch GmbH, Stuttgart, Germany
[2]Ruhr-University Bochum, Germany

ABSTRACT
Mechatronic products are realized by combinations of different engineering domains, namely mechanics, electronics and software. Therefore, a major step in the design process of mechatronic systems is the partitioning of the system's functions among these domains. This design step, also referred to as domain allocation, is an essential problem in mechatronic design. Domain allocation is performed in the conceptual design phase. Usually experts for mechanics, electronics and software are working in cooperation in order to achieve an advantageous allocation with respect to the performance of the overall system. During the design process, different levels of abstraction are used in order to describe the system to be developed. Especially in mechatronic design, the complexity of the system leads to the fact, that certain aspects of the solution are only known on an abstract level, while others already can be described on a higher level of concretization. Particularly during domain allocation, functions, solution principles and components have to be considered in parallel within the same product model. This paper presents a heterogeneous modelling approach which allows the combination of different levels of abstraction. Concrete and abstract model elements can be connected and their interaction can be described, which enables engineers from different domains to understand the system and to modify the solution in a flexible manner. The modelling approach has been implemented as a computer tool in order to test its applicability in practice. The methods are verified by the development of a mechatronic system, namely a snake-like robot.

Keywords: mechatronics, design process, domain allocation, heterogeneous modelling, snake- like robot

1 INTRODUCTION

The development of mechatronic systems is an extremely complex process due to the required integration of different technologies into a complete technical system. Domain allocation represents a central aspect of the mechatronic development process [1]. In this context a mechatronic system is subdivided into subsystems based on mechanics, electronics and information technology, whose interaction has to be realized in an appropriate manner. The objective consists in the optimization of a system's properties with respect to the product requirements. The optimal interaction of mechanics, electronics and information technology can only be achieved by a cross-domain approach which makes an examination of the solution space as comprehensive as possible, in particular in the early phases of the development process. In order to satisfy these demands, a methodology for domain allocation has been developed [5], which comprises a flexible modelling approach presented in this contribution.

This paper proposes a heterogeneous modelling approach which enables the product designer to describe a mechatronic system by coupling system elements described on arbitrary levels of abstraction. Furthermore, the transfer of the introduced concept into a computer-aided tool is presented. As an example, the application of the methodology for domain allocation as well as the heterogeneous modelling approach are illustrated by means of a development process carried out for a multifunctional movement system, namely a snake-like robot.

2 PARTITIONING IN MECHATRONICS

The overall functionality of mechatronic systems is fulfilled by the interaction of solution principles of different domains. In general, numerous possible combinations of solutions principles based on different domains exist to realize the required product functions, which lead to a variety of different domain structures. During the development process of a mechatronic system, one has to determine in which way the different domains shall interact to obtain a system's required properties. This process is referred to as partitioning, domain allocation [2] or technology allocation [1].

- The assignment of a function or a group of functions of a system to a certain domain (mechanics, electronics, information technology etc.), or a combination of these domains is called *functional partitioning* of a mechatronic system. Also the relations between the sub-functions are assigned to the different domains.
- *Spatial partitioning* covers the geometric order and structural grouping of system elements within a mechatronic system, particularly taking into account the domains used for its realization.

The modular design of mechatronic systems and the realization of a system's functionality by an interaction of mechatronic modules are of major importance in the context of partitioning. For example, an essential aim can consist in achieving a system's degree of modularization as high as possible which leads to extensible as well as flexibly usable modules. Furthermore, the structuring of a system with respect to information processing also has to be taken into account during partitioning. It has to be determined how the functions realized in information technology are split among the individual modules and which hierarchical and non-hierarchical relations shall be between them.

The structure of a mechatronic system with regard to the domains involved is presented in Figure 1. The control engineering domain is not explicitly contained in the illustration since the associated functions are realized by one or several of the other domains within the system. The basic system is primarily based on the mechanical domain which is completed by e.g. thermodynamic, optic or acoustic working principles. The basic system's state variables are influenced by actuators whose energy converters form the interface between the mechanical and the electrical subsystems. In combination with power electronics the basic system represents the *energy dominated* part of the mechatronic system.

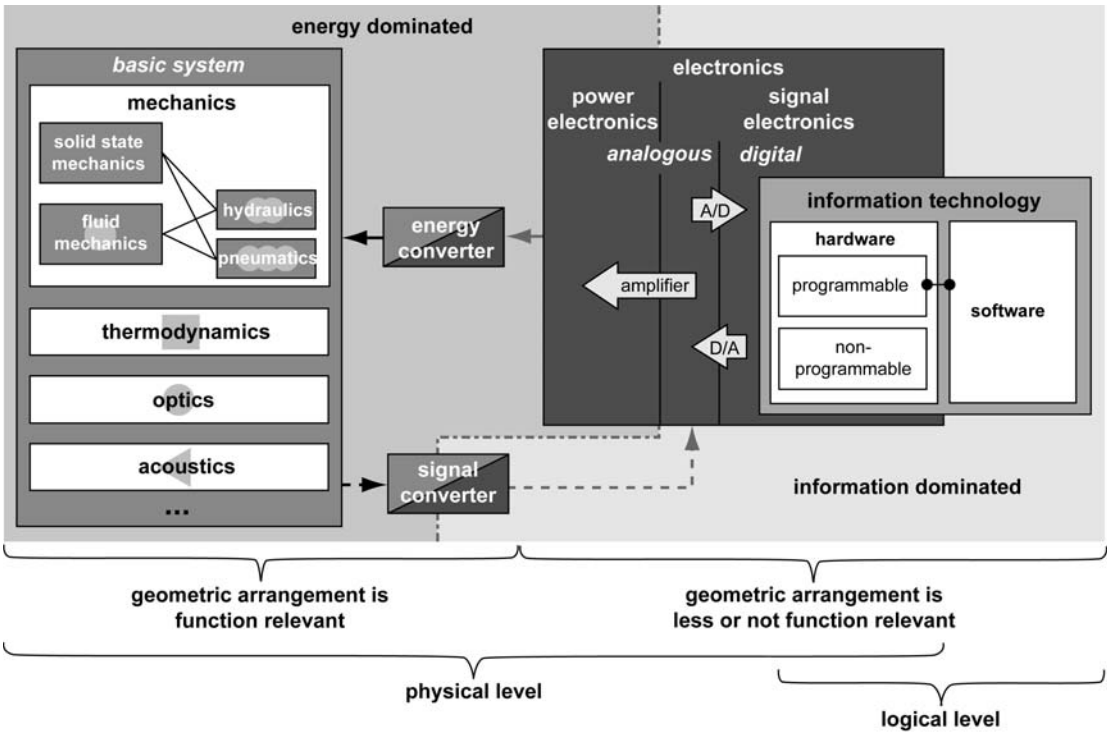

Figure 1. Interaction of domains in a mechatronic system

Both, the signal transducers of the sensors as well as the electronic power amplifiers serve as interfaces between the energy and *information dominated* subsystems. While the signal transducers deliver electrical signals as raw input quantities for analogue electronics, the amplifiers provide the electrical power for the energy converters of the actuators.

From the viewpoint of functional partitioning, actuators as well as sensors represent interfaces between different domains [3]. With regard to the actuators this interface is located between domains of the energy dominated part of a mechatronic system. By this means, an influence of a mechanical subsystem by an electrical subsystem can be achieved.

Sensors represent interfaces between energy dominated and information dominated subsystems as well as between different domains. The selection of sensors which are required in a mechatronic system for function fulfilment is therefore directly influenced by the domain structure. For example the number of sensors increases with the number of movements to be controlled independently so that the effort for sensors grows with an increasing substitution of mechanical couplings by those realized by information technology. On the other hand, a specific variation of the sensory interfaces can lead to alternative ideas for the partitioning of mechatronic systems.

In order to achieve a system’s global optimum, the analysis of information processing in the context of functional domain allocation firstly concentrates on the question, which of the involved functions should be realized by an energy dominated and which should be assigned to an information dominated subsystem. Furthermore, a division must be carried out between analogue and digital sub-functions within the information processing subsystem. An assignment of the functions to the electrical or information technology domain is carried out.

An aspect important to the spatial partitioning consists in the different relevance of the geometric order of components within the different domains. In systems in which non-electrical variables take in a central role for function fulfilment, the geometric arrangement of the respective elements and their relative location to each other are function relevant. This becomes particularly clear on the example of the mechanical domain where the desired movements can be realized only if component geometries and dimensions are coordinated with each other in a suitable way and furthermore the components are coupled with each other by connections which show the required degrees of freedom. The geometric arrangement is, however, of importance also for thermal, optic or acoustic subsystems. Examples are thermally activated energy converters such as shape memory alloys or bimetals, optical angle detectors as well as distance sensors based on supersonic whose functions are strongly influenced by the spatial arrangement of the components involved.

In electrical subsystems the spatial partitioning of individual elements as well as that of the circuits based on them is fundamentally less critical. On the one hand, this is due to the relatively simple transmission of electrical power and signals, on the other hand due to the extensive encapsulation of the physical effects inside of discrete components. These components have well-defined interfaces in form of electrical ports which are connected to each other for the realization of the desired functions. These circuits can be relocated largely flexible so that the arrangement of the components or wiring can be varied.

2.1 Integration of domain allocation into mechatronic product development

For the structuring of development processes for technical systems different procedure models exist which describe different gradual working steps as well as developmental stages. The integration of partitioning of mechatronic systems into the development process is carried out using the procedure model described in VDI guideline 2221 [4]. This guideline is in principle applicable for a large spectrum of products and also takes into account the development of products within the domains of mechanical engineering, process and precision engineering as well as software engineering.

Starting point for an iterative development process is the formulation of a task. On this basis the requirements on the product to be developed are determined in the first work step. In the next step, functions which have to be fulfilled by the product are defined. As a rule, at first the complete function of the system to be developed is described and essential input and output quantities are determined. This overall function is divided into sub-functions during the following detailing steps. From the arrangement and combination of the sub-functions a function structure arises.

In the next step a search for solution principles is carried out for the identified functions which are then connected to working structures. A solution principle describes the fundamental realization of one or more functions. Within mechatronic systems both physical as well as non-physical effects based on information technology are used to create the working principles.

In the following design steps, a structuring of the principle solution into realizable modules is carried out which are first described as a rough concept and then concretized by detailed creative specifications. The last design step within the development process comprises the preparation of the product documentation.
The integration of domain allocation into the product development process outlined before is presented in Figure 2. While the functional partitioning is carried out in a very early phase in which the system to be developed is described on an abstract level, the spatial partitioning extends over several different development phases with different concretization degrees.

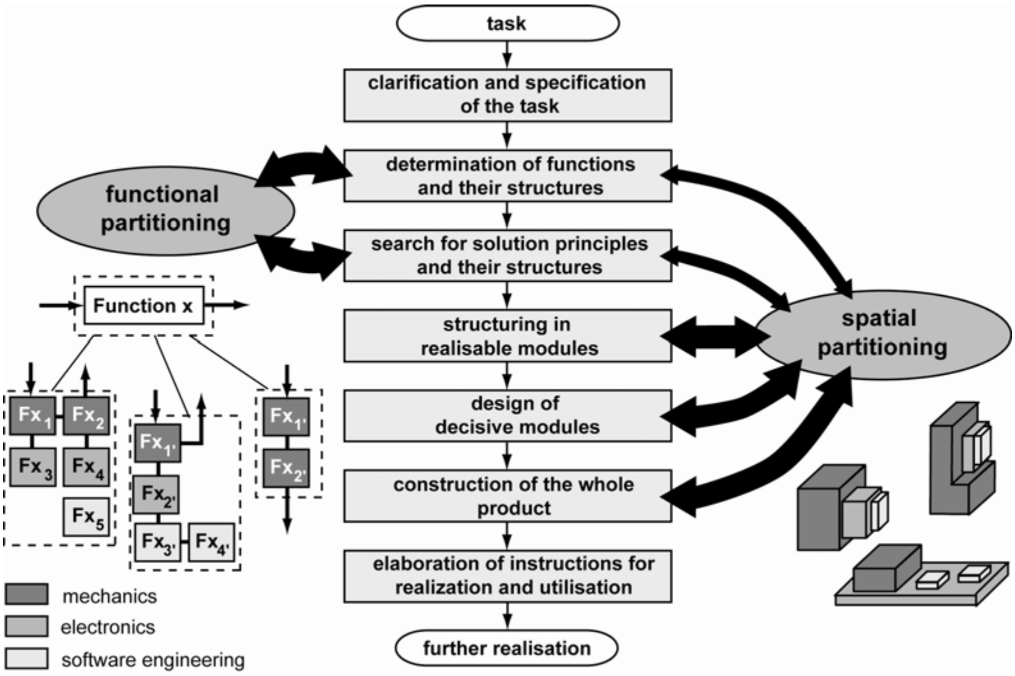

Figure 2. Integration of functional and spatial partitioning into the development process

As indicated by the arrows, there is a bidirectional dependence between partitioning and the contents and results of the working steps involved. For example, an adaptation of the *domain structure* can be necessary due to a variation of the function structure in the context of functional partitioning. On the other hand a modification of the domain structure may lead to additional functions which then have to be integrated into the function structure. Furthermore, during the search for solution principles it can happen that no suitable solutions based on the domain structure can be generated due to faults during partitioning or a disadvantageous level of detail chosen for the product model. Also in these cases a further execution of the functional partitioning is required. The situation described can be applied analogously to spatial partitioning where the necessity can arise to optimize the spatial arrangement of the domains, e.g. due to certain shape requirements.
As a part of the mechatronic development process, also partitioning is subject to iterations. Therefore the domain structures must flexibly be expandable and allow variations. Since particularly the spatial partitioning is carried out in different phases of the development process, a consideration of different concretization degrees in the representation of the domain structure as well as in the methodical support of the process of domain allocation is required.

2.2 Analysis of existing methods and tools for the support of domain allocation

Existing attempts for the support of domain allocation in mechatronic product developments have been examined in a comprehensive analysis [5]. At first, process-oriented approaches have been considered, which aim on the provision of procedure models as well as methods for mechatronic product development. After that, approaches which aim on structuring and modelling of mechatronic systems have been explored. The state of research has been analyzed systematically and evaluated.
For the subject of *system structuring and modelling* [6 – 16], different evaluation criteria have been defined, namely the fulfilment degrees with regard to the modelling of structural relations, the modelling of spatial and geometric features, the modelling on a functional level, the modelling on the

level of working principles, the modelling on component level, the simulation of systems, formal modelling, informal modelling as well as the representation of domain structures and -interfaces.

The analysis results show that several approaches have a quite high fulfilment degree in the area of modelling structural relations. Nevertheless, representable relations do not cover all connections between system elements which must be taken into account during the development of mechatronic systems.

Although geometric and spatial aspects play an essential role in the development process, their representation is supported only by few of the concepts considered. Among other things this is due to a high abstraction degree of modelling used in most of the approaches. Thus, system elements are frequently represented by simple graphic constructs which permit merely the construction of two-dimensional structures without a direct reference to the real geometric arrangement of the elements. The combination of formal and informal descriptions is only included in a few approaches. No approach is explicitly used for the visualization of domain structures although it is possible in some cases to assign information about the underlying domain to the system elements.

However, it can be noticed that a closed concept which can be used for the modelling of mechatronic systems and which supports both functional as well as spatial domain allocation does not exist. Nevertheless, different aspects which are of importance for a system's partitioning are considered by some of the analyzed approaches.

3 A HETEROGENEOUS MODELLING APPROACH AS AN ESSENTIAL ELEMENT OF A METHODOLOGY FOR DOMAIN ALLOCATION

In this chapter the developed modelling approach is introduced which is suitable for the partitioning of mechatronic systems in particular but is also transferable to other areas of product development. During the development of this approach, requirements have been determined in a consequent manner [5]. Their fulfilment assures efficient applicability of the developed tool.

3.1 Model elements and their interfaces

The modelling of a mechatronic system to be developed is carried out by the help of model elements. All model elements (system, disturbance and context elements) together with their relations describe a mechatronic system which cannot be further subdivided under consideration of the chosen concretization level. The structure of a *system element* is shown in Figure 3 representatively for the different model elements.

For the description of system elements, different formal and informal concepts are chosen. With the help of verbal descriptions, the developer has the possibility of representing arbitrary aspects of a system element on a *non-formal level*. For instance, the desired transfer characteristics of a function can be described qualitatively or special features of a working principle or a component can be explained. Another informal mean for system description consists in the illustration of a system element using roughly sketched representations. This representation form is particularly an essential aid for the development, analysis and documentation of solutions in the early phases of the development process. A main advantage in comparison to more formal representations such as CAD models is the capability of representing also incomplete, blurred concepts on an arbitrary level of abstraction. This enables the developer to model the product without unnecessary concretizations and to concentrate on the essential concept components instead. In principle, with the help of sketches mechanical, electric as well as information technology solution concepts can be visualized so that they also can serve as a communication basis for experts of different disciplines.

A description of the behaviour of system elements on a *formal level* can be achieved by the help of diagrams, equations or algorithms. With diagrams, both continuous and discrete processes can be modelled. For example, interpolated function graphs can be used to model the desired transient response if a description on the mathematical level by equations is time-consuming or not required. For the description of system elements with discrete behaviour, state diagrams are particularly suitable. Besides graphic modelling of discrete system behaviour, textual descriptions based on higher programming languages are useful especially on high concretization level. Thus, even complex algorithms can be represented, so that the behaviour of the information technology system components can be described close to its subsequent realization.

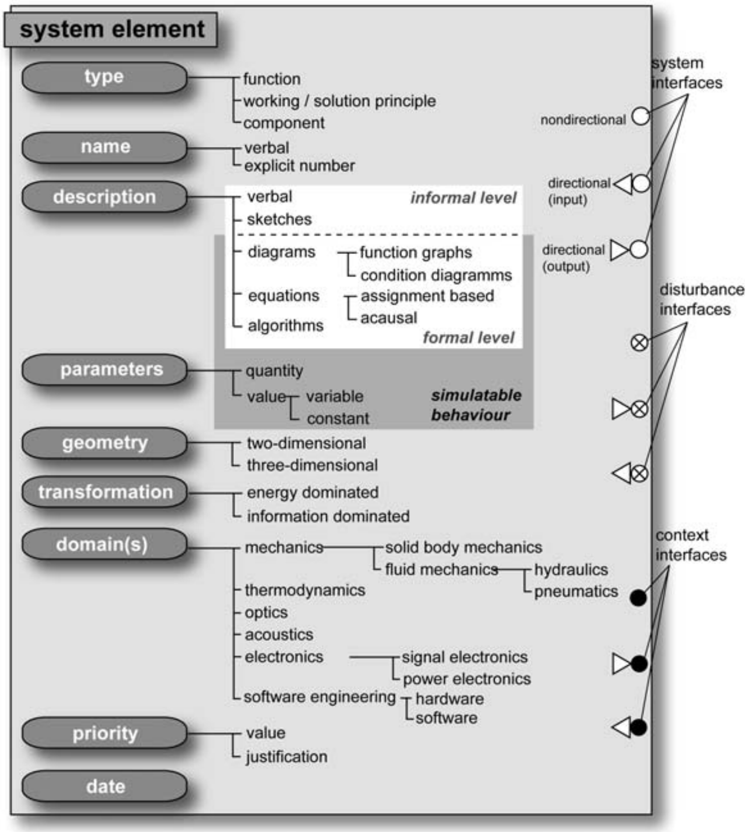

Figure 3. Structure of system elements

To make the system elements utilizable for modelling in the context of domain allocation, the assignment to the energy or information dominated subsystem and a specification of the domains underlying the elements are necessary. After the assignment of the domains has been carried out during functional partitioning, the resulting structures can be analyzed with respect to the interaction of the different disciplines. Inconsistencies can be found within the coupling of elements or optimization potentials for the domain structure may be recognized in functional and spatial regard through this, for example. Furthermore, certain requirements or restrictions already arise from a domain specification which are then connected to the system elements in form of *context elements*.

The coupling of system elements is carried out via system interfaces which can be both nondirectional and directional with regard to their assignment characteristics. Nondirectional interfaces are particularly suitable for the representation of physical couplings between energy dominated system elements. Directional system interfaces by which the exchanged quantities are transferred non-reactively represent inputs or outputs of the system element and are suitable for modelling information-dominated subsystems primarily.

A system element also can have disturbance interfaces besides the system interfaces. These serve for the connection of *disturbance elements* (e.g. sources of temperature, humidity or magnetic field) and for the coupling of system elements with potential disturbance effects. As in the case of the system interfaces, directional and nondirectional transmission characteristics can also be modelled here to take into account the influence direction of the disturbing effects.

Furthermore, system elements can have interfaces for the coupling with context elements. These permit to include the relations of the system elements to the related relevant aspects of the development context, such as *requirements*, *resources* or *restrictions*. Context elements are not a part of the system itself and they do not show any functional or physical behaviour. Nevertheless they are important due to the organizational and administrative information which they represent.

3.2 Relations between model elements

Relations are used to connect model elements to each other. Physical couplings which connect the state variables of system elements to each other serve for the modelling of physical connections. In

principle, *non-directional* and *directional relations* can be distinguished. While influences are possible in both directions for nondirectional couplings and also reactions of the system elements are therefore included, this is not the case for directional relations.
Since the behaviour of energy-dominated couplings of system elements can only be modelled realistically if the bidirectional character is considered, primarily non-directional connections are used in energy-dominated subsystems. Information dominated couplings usually can be described as non-reactive connections between inputs and outputs of information-processing system elements. Directional relations are therefore preferably used here.
Figure 4 shows how energy and information dominated relations between system elements can be modelled by directional and non-directional couplings on different abstraction levels.

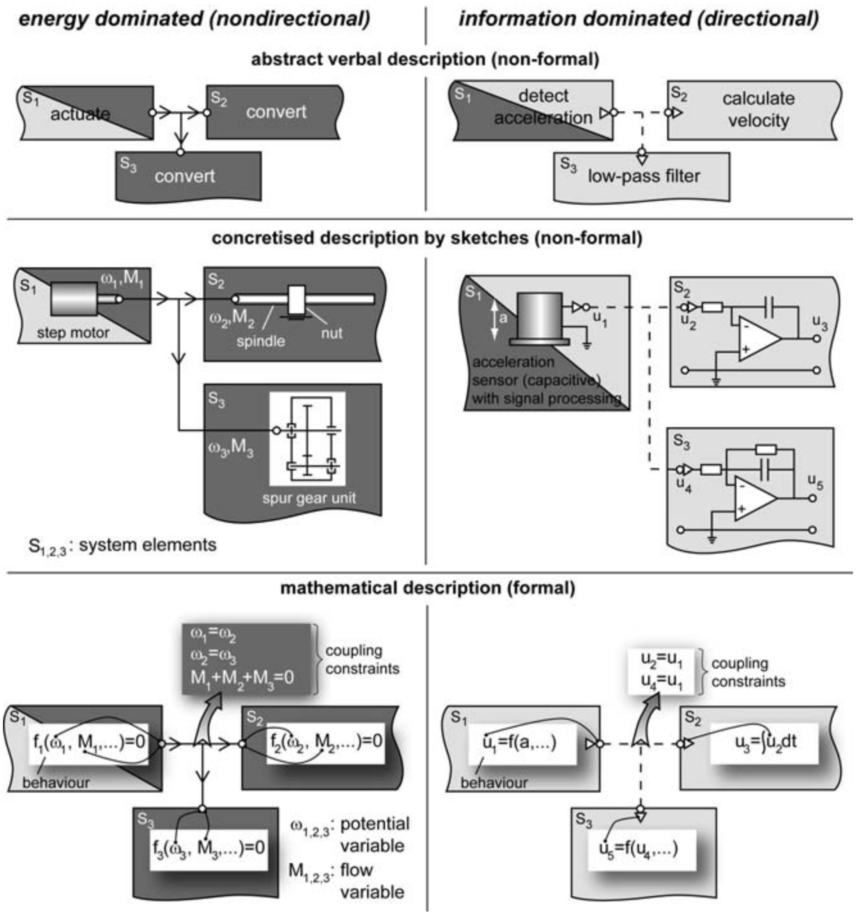

Figure 4. Examples for modelling of energy and information dominated relations between system elements

By the help of a verbal (informal) description of system elements, the nature of the physical quantities representing the energy or flow of information is defined like in the upper part of Figure 4. The quantities are not specified in an early developmental phase, since the solution principles for the system elements have not been selected yet. It is merely known that an energy or information dominated coupling of the elements must be realized to fulfil the system’s function. In the case of an energy dominated system, the modelling of the relations is carried out via nondirectional relations in which the direction of the power flow can be set. The information dominated connections are modelled, however, by directional relations in which system interfaces are used as inputs and outputs.
A concretised (informal) description of system elements is achieved by sketches as a representation of solution principles selected for the realization of functions. The physical quantities as well as signals transferred between the elements are known now and can be assigned to the system interfaces so that the consistency of the model can already be checked in this stage.
The physical behaviour of system elements is described on a mathematical level and the quantities involved in the relations are assigned to the corresponding interfaces. This is represented by a mathematical (formal) description, shown in the lower part of Figure 4. In connection with the

equations for the description of a system element's behaviour, a simulation of the energy dominated system can be carried out. By the directional relations between the information dominated system elements, the equality of the interface variables is enforced and the calculation order fixed.

3.3 Model representation

For the representation of models of mechatronic systems and their efficient use suitable representation forms must be available for the stored information. Depending on the aim of examination, different types of model representation can be useful. While in the context of functional partitioning particularly the representation of the connections between product functions and the assigned fulfilment domains are of importance, spatial and geometric aspects play an important role in later phases of development. Therefore, besides purely schematic representations of the product which show the couplings between the system elements in two-dimensional structures, detailed three-dimensional model views also have to be provided. It can make sense to choose different abstraction degrees in a single representation according to the heterogeneous modelling approach for the model elements. Often concrete descriptions for certain subsystems already exist on component level or on the level of working principles, while for other parts of the product only a rough functional description is available.
Using the information stored in the model elements and their relations it is possible, to flexibly adapt the representation form to the developer's needs and to represent only context relevant aspects of a concept. Kinematical couplings between mechanical components can be represented both on a realization near level by the coupling of three-dimensional bodies as well as on an abstract level by the help of two-dimensional symbols and corresponding connecting lines. Although in the second case some available information like the geometry of the bodies and their spatial arrangement are not taken into account, the analysis of fundamental connections can be made easier especially for complex products with many components in the conceptual design phase.

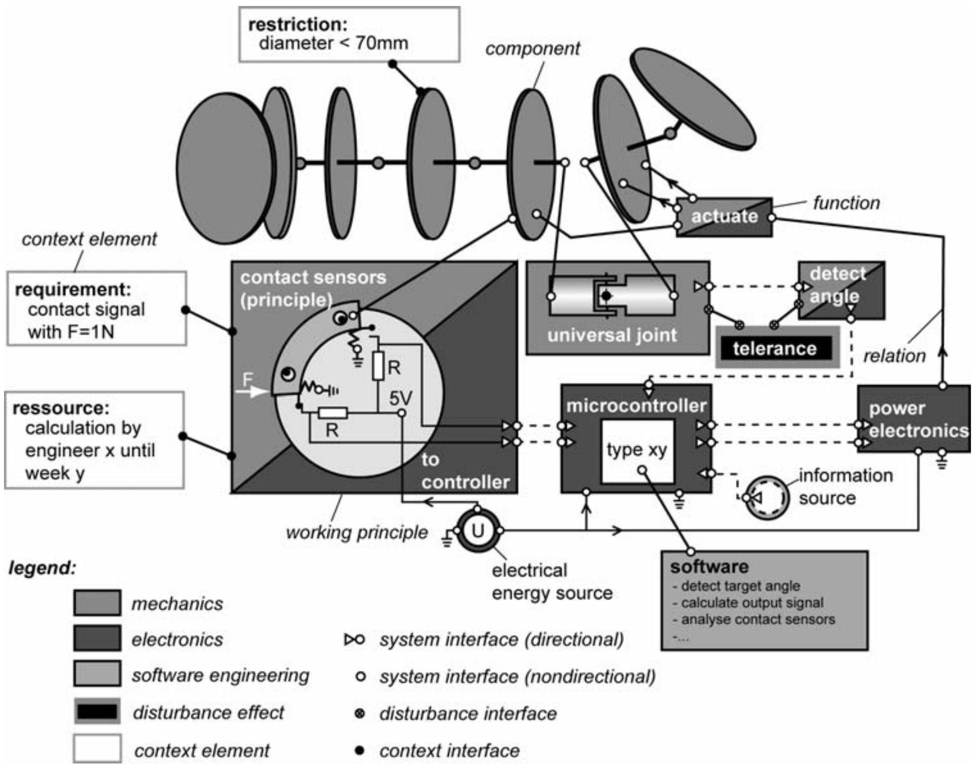

Figure 5. Model representation forms for a developed snake-like robot

Figure 5 shows exemplarily different representation forms of system elements in a concept model for a snake-like robot developed at the Institute of Engineering Design [5,17], which also has been used as an example for the evaluation of the developed modelling approach (see chapter 4). Included are system elements (e.g. universal joint, microcontroller, software), disturbance elements (e.g. joint tolerance) and context elements (e.g. requirement: contact signal, restriction: diameter), at different abstraction levels coupled by directional and non-directional relations. The representation already shows the functional and spatial partitioning of the snake-like robot carried out. This type of model representation provides a holistic overview of the system to the developers of all disciplines in the concept phase. In particular, first analyses through compatibility checks of energy and signal flows regarding the consistency are possible in this stage.

3.4 Software prototype implementing the heterogeneous modelling approach

The prototypic transfer of the heterogeneous modelling approach to a software tool has been carried out using the object-oriented methodology [18]. This is particularly suitable for building the domain structures since the model elements and the relations between them can be interpreted as objects. In these objects data required for the description of the model elements are stored encapsulated in attributes. The access to these data as well as their manipulation is carried out via the methods defined in the underlying classes. The object orientation allows a realistic description of the product concept since the model of the mechatronic system consists of single elements which can be understood as objects that are coupled to each other by their interfaces. The properties of the elements can be modelled as attributes. Objects produced once can be stored, duplicated and used in other models again. Thus, it is possible to reduce the effort for the model design and to configure subsystems of available model elements with protected properties. The reliability of the model can be increased by this, too.

The application [3, 5 and 19] has been realized by use of the programming language Java which implicates the advantages of platform independence, strict object orientation and free availability. For the realization of three-dimensional shapes of a concept, Java3D is used which extends the Java platform by an efficient class library for the visualization of virtual worlds. Furthermore, the development of the software tool has been based on a model-view-controller-architecture. The developed software is marked by a structural separation of classes for data storage, presentation and processing which are stored in different libraries. This makes it easier to carry out extensions and changes of the software.

Figure 6 shows the user interface of the software tool. The model editor serves as a central user interface for a flexible development as well as visualisation of mechatronic systems using the heterogeneous modelling approach.

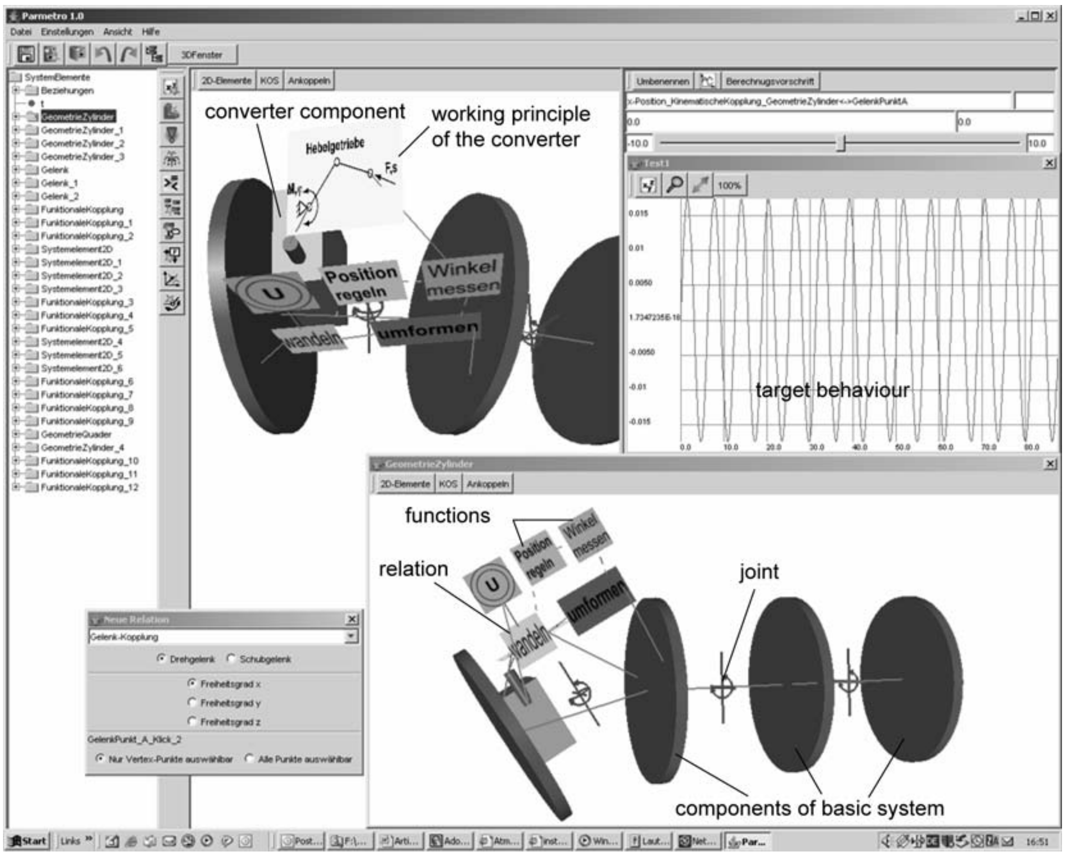

Figure 6. User interface of the modelling application

The model elements of different abstraction levels can be coupled with each other and appear simultaneously in the same view. The model shown here represents a part of the snake-like robot. The view of the three-dimensional model can be varied by the help of rotation, translatory movement and zoom functions. A direct access to the model elements and relations is possible by mouse click, which can be included as well as removed. An overview of the produced model elements is shown in the model element administration under consideration of the hierarchy relations in form of a tree structure. It is further possible to limit the elements contained in this structure using a filter function depending on a desired view so that it is possible to represent only the system or context elements or to carry out e.g. a choice using the structure criterion "domain". The access to the program functions is performed in the main menu or by the use of icons. The interaction with the developer is realized via corresponding windows which enable the input and output of model-related data.

4 EVALUATION OF THE HETEROGENEOUS MODELLING APPROACH EXEMPLIFIED BY A SNAKE-LIKE ROBOT

As an evaluation example a multifunctional movement system has been developed [5 and 20]. It has been inspired by the biological archetype of a snake. Despite their comparatively simple body form snakes are able to execute a variety of complex movement forms. For example different ways of locomotion like lateral undulation, side winding, climbing or swimming are possible. The various movement forms are made possible by the structure of the snake body. It consists of uniform segments which have a simple construction and a restricted number of degrees of freedom. A flexible complete system which can execute the different movements arises from the coupling of a variety of these segments in connection with a complex control of the muscle apparatus.

The derivation of technical solutions from the principles of the natural snake and the use of the potentials according to bionics arising from it establish a wide application range for mechatronic systems with a snake-like design. So the use of snake-like robots of different scaling is imaginable for the reconnaissance of hardly accessible surroundings or for the inspection and maintenance of pipes, for example. A use of such systems as manipulators in an industrial or domestic environment or in the context of medical engineering is in principle also possible due to the high flexibility of movement forms. For some years the development of snake robots has therefore been addressed in different research projects which devote themselves to this topic against the background of different goals and application scenarios [21 – 23].

The result of the development process is shown in Figure 7. The snake-like robot has been build up from single modules designed by the use of the proposed heterogeneous modelling approach. The modules are connected mechanically over bayonet locks which simplify assembly and disassembly of the complete system and make the variation of its configuration easier. The circuit boards fulfil also mechanical functions besides the electrical functions and therefore represent multi-domain elements, which could be determined in an early development phase due to the partitioning methods in combination with the modelling approach. Besides fulfilling the electrical functions, the board serve as carriers for the universal joint and the contact sensors and transfer the forces produced by the actors at the same time.

Every module is equipped with a microcontroller which receives the desired angles to be adjusted by the actuators and transforms them into corresponding PWM signals which are used to activate the servomotors. Every module has two voltage regulators of which one serves for the low-power supply of the information-processing components while the other provides the electrical power for the actuators and is realized as a switching regulator in order to reduce dissipation. By the decentralized voltage regulation the required power can be transferred to the segments on a relatively high voltage level which leads to lower currents and power loss in the supply circuits. The calculation of the desired positions for the actuators is carried out centrally by a PC as well as the specification of the required snake-curve and the determination of the segment angles. The desired movement is caused by the fact that the snake-curve is varied in short intervals so that harmonic movement processes arise. The PC transmits the calculated values by the help of a TCP/IP interface to the embedded controller functioning as a bus master which passes the data via an I2C-bus to the individual modules. The communication can be carried out in two directions so that the information gained by evaluation of the contact sensors can be transmitted for instance, too. This permits the system to react to objects in its environment e.g. by adaptation of the movement form.

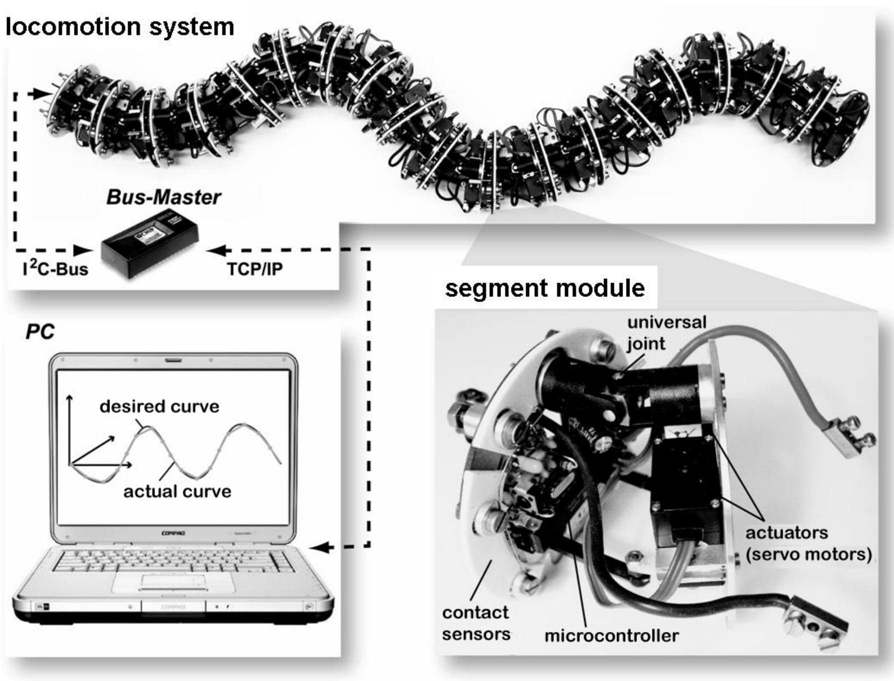

Figure 7. Realized prototype of the snake-like robot

With this evaluation example it has been shown that the heterogeneous modelling approach supports particularly domain allocation in the context of mechatronic product developments at a functional and spatial level. The approach stands out due to the possibility of a combination of different levels of abstraction within the same model and permits a partial concretization of the system. The information relevant for the partitioning is stored in the model and can be used to support variation of the domain structure. In addition to system elements, disturbance effects and relations also context elements are used, which enable the modelling of organizational and administrative aspects like requirements and restrictions.

REFERENCES

[1] Buur J. *A Theoretical Approach to Mechatronics Design*, doctoral thesis, Technical University of Denmark, Lyngby, 1990.

[2] Welp E.G. and Jansen S. Domain Allocation in Mechatronic Products, *Proceedings of the 8th International Design Conference (DESIGN 2004)*, Vol. 2, Dubrovnik, pp. 1349 – 1354, 2004.

[3] Jansen S. and Welp E.G. Model-Based Design of Actuation Concepts: A Support for Domain-Allocation in Mechatronics, In: *Proceedings of the 15th International Conference on Engineering Design (ICED)*, Melbourne, Paper No. 218.49, 2005.

[4] VDI. *VDI-Richtlinie 2221: Methodik zum Entwickeln und Konstruieren technischer Systeme und Produkte*, Berlin: Beuth Verlag, 1993.

[5] Jansen S. *Eine Methodik zur modellbasierten Partitionierung mechatronischer Systeme*, doctoral thesis, Ruhr-University Bochum, Institute of Engineering Design, 2007.

[6] Kallmeyer F. *Eine Methode zur Modellierung prinzipieller Lösungen mechatronischer Systeme*, doctoral thesis, University Paderborn, 1998.

[7] Hahn M. *OMD – Ein Objektmodell für den Mechatronikentwurf*, Düsseldorf, VDI-Verlag, 1999.

[8] Langdon P.M., Bracewell R.H., Duffy M.J.W. and Sharper J.E.E. An integrated design platform for the conceptual design of complex mechatronic systems, In: *Proceedings of the 2nd International Conference on Mechatronics and Machine Vision in Practice*, City University of Hong-Kong, pp. 333 – 338, 1995.

[9] Counsell J., Porter I., Dawson D. and Duffy M. Schemebuilder: computer aided knowledge based design of mechatronic systems, In: *Assembly Automation*, Vol. 19, No. 2, pp. 129 – 138, 1999.

[10] Van Amerogen J. Mechatronic design, In: *Mechatronics*, Vol. 13, No. 10, pp. 1045 – 1066, 2003.

[11] De Vries T.J.A. *Conceptual Design of controlled electro-mechanical systems – a modeling perspective*, doctoral thesis, University of Twente, Enschede, 1994.
[12] Stacey M., Sharp H., Petre M., Rzevski G. and Buckland R. A representation scheme to support conceptual design of mechatronic systems, In: Gero J.S., Sudweeks F. (editors): *Artificial Intelligence in Design 1996*, Dordrecht: Kluwer Academic Publishers, pp. 583 – 602, 1996.
[13] Chen L., Jayram M. and Xi J.F. A new functional representation scheme for conceptual modeling of mechatronic systems, In: *Proceedings of DETC'02 ASME 2002 Design Engineering Technical Conferences and Computer and Information in Engineering Conference*, Montreal, Paper No. DETC2002/DTM-34012, 2002.
[14] Petrik O. and Szász G. The inside structure model of mechatronic devices, In: *Mechatronics*, Vol. 3, No. 2, pp. 127 – 138, 1993.
[15] Mrozek Z. Computer aided design of mechatronic systems, In: *International Journal of Applied Mathematics and Computer Science (AMCS)*, Vol. 13, No. 2, pp. 255 – 267, 2003.
[16] West A.A., Harrison R., Wright C.D. and Carrot A.J. The visualization of control logic and physical machine elements within an integrated machine design and control environment, In: *Mechatronics*, Vol. 10, No. 6, pp. 669 – 698, 2003.
[17] Jansen S. and Welp E.G. Modellbasierte Partitionierung mechatronischer Systeme am Beispiel eines multifunktionalen Bewegungssystems, In: *4. Paderborner Workshop „Entwurf mechatronischer Systeme"*, Paderborn: HNI, 2006.
[18] Rumbaugh J., Blaha M., Premerlani W., Eddy F. and Lorensen W. *Objektorientiertes Modellieren und Entwerfen*, München: Hanser Verlag, 1993.
[19] Jansen S. and Welp E.G. Ein Werkzeug zur modellbasierten Partitionierung mechatronischer Systeme, In: *Mechatronik 2005 – Innovative Produktentwicklung*, VDI-Berichte 1892.1, Düsseldorf, VDI Verlag, pp. 273 – 293, 2005.
[20] Exhibition of the locomotion system at the "ThyssenKrupp Ideenpark", Hannover, 2006. www.zukunft-technik-entdecken.de → Ideenpark → Exponate → Kreativität → "Leben erleichtern" → „Der Schlangenroboter – Multifunktionalität durch Mechatronik"
[21] Miller G.S.P. Snake Robots for Search and Rescue, In: Ayers J., Davis J.L. and Rudolph A. (editors): *Neurotechnology for Biomimetic Robots*, Cambridge: MIT Press, 2002.
[22] Dowling K.J. *Limbless Locomotion: Learning to Crawl with a Snake Robot*, doctoral thesis, The Robotics Institute, Carnegie Mellon University, Pittsburgh, 1997.
[23] Klaassen B. and Paap K.L. GMD-SNAKE2: A snake-like robot driven by wheels and a method for motion control, In: *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 3014 – 3019, 1999.

Contact: Ewald G. Welp
Ruhr-University Bochum
Institute of Engineering Design
Universitätsstr. 150
44801 Bochum
Germany
Tel.: Int +49 234 32-23637
Fax: Int +49 234 32-14159
E-mail: welp@lmk.rub.de
URL: http://www.lmk.ruhr-uni-bochum.de/